LM-GA: A Novel IDS with AES and Machine Learning Architecture for Enhanced Cloud Storage Security

[1]T Thilagam and [2]R Aruna
[1,2]VelTech Rangarajan Dr. Sagunthala R & D Institute of Science and Technology, TamilNadu, India.
[1]thilaka28@gmail.com, [2]drraruna@veltech.edu.in

Correspondence should be addressed to T Thilagam : thilaka28@gmail.com.

Article Info
Journal of Machine and Computing (http://anapub.co.ke/journals/jmc/jmc.html)
Doi: https://doi.org/10.53759/7669/jmc202303008
Received 18 August 2022; Revised from 12 December 2022; Accepted 30 December 2022.
Available online 05 April 2023.

Abstract - Cloud Computing (CC) is a relatively new technology that allows for widespread access and storage on the internet. Despite its low cost and numerous benefits, cloud technology still confronts several obstacles, including data loss, quality concerns, and data security like recurring hacking. The security of data stored in the cloud has become a major worry for both Cloud Service Providers (CSPs) and users. As a result, a powerful Intrusion Detection System (IDS) must be set up to detect and prevent possible cloud threats at an early stage. Intending to develop a novel IDS system, this paper introduces a new optimization concept named Lion Mutated-Genetic Algorithm (LM-GA) with the hybridization of Machine Learning (ML) algorithms such as Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM). Initially, the input text data is preprocessed and balanced to avoid redundancy and vague data. The preprocessed data is then subjected to the hybrid Deep Learning (DL) models namely the CNN-LSTM model to get the IDS output. Now, the intruded are discarded and non-intruded data are secured using Advanced Encryption Standard (AES) encryption model. Besides, the optimal key selection is done by the proposed LM-GA model and the cipher text is further secured via the steganography approach. NSL-KDD and UNSW-NB15 are the datasets used to verify the performance of the proposed LM-GA-based IDS in terms of average intrusion detection rate, accuracy, precision, recall, and F-Score.

Keywords – Cloud Security, Intrusion Detection System, Lion Mutated-Genetic Algorithm, Convolutional Neural Network, Long Short-Term Memory, Hybrid Deep Learning.

I. INTRODUCTION

Cloud Computing environment enabled storage and sharing of a wide range of digital data through the internet, encompassing text, picture, video, audio, and so on [9][10]. The number of people accessing the internet has dramatically increased recently. To meet the needs of the users, a system with broad access and storage is necessary. Users can more efficiently store information in the cloud and share it online [11]. People of cloud systems can access and distribute digital data across numerous users all over the world. Additionally, cloud technology enables users to import and export data via web object storage, even if this data may contain sensitive information like employee profiles or corporate data [12]. The security of data shared via the internet is still susceptible to attacks like brute force attacks and occlusion attacks, despite the cloud's abundance of services and vast storage capacity [13][14]. Additionally, hacking is another risk that can result in data loss or damage during data transport and storage [15].

Numerous studies have been conducted to improve secure data sharing using key generation methods in order to address the problems with cloud computing [16] [1]. However, it has problems with authentication and trust between the data sender and receiver [17]. The key cryptosystem occasionally takes a long time and may result in mistakes. Later, IDS is introduced for protected cloud data exchange [23]. Thereby, an efficient IDS should be able to handle key functions like data extraction, interpretation, intrusion detection, and reporting [19]. To accomplish these goals, the traditional IDS approach frequently employs statistical modeling, AdaBoost algorithms, Hidden Markov Model (HMM), CNN, and Back Propagation-Neural Network (BP-NN) [2][20][21][22]. However, these methodologies did not cope with the modern cloud architecture [3]. The issues in performance continue by means of computational complexity as well as high execution time. The most advanced IDS system, which relied on shallow ML approaches, had low accuracy, a greater false alarm rate, and a longer reaction time [18].

With the intention of overcoming the limitations, researchers introduced numerous optimization concepts, for optimizing the key management system while key generation and data transmission [35]. AES codes are established using unique structures which can answer various problems at the same time. Generally, AES functions similar to substitution–permutation network design idea and is effective in both software and hardware [4]. However, the limitations like overly simplistic algebraic structure, problematic in implementing the software, and each block is encrypted in exactly a similar manner. Furthermore, when it comes to performance and security, AES in counter mode is hard to execute in software [5] [6]. Due to these reasons, optimization in the key management approach is introduced in the AES encryption algorithm. For this purpose, the paper introduces a novel IDS-based encryption model using an LM-GA scheme to generate optimal keys are used encryption and decryption processes. The key contribution of the paper is as follows.

- First, the attack classification is performed on the input intrusion data. The classification is performed through the utilization of the Neural Network (NN)technique namely the LSTM network along with CNN.
- A hybridization of LSTM is utilised for training since CNN can cause high error rate due to enormous numbers of packets in the CNN. Thus, a high classification rate and a low error rate can be attained.
- The IDS monitors the arrival of new data for intrusions, attacks, and other problems before responding with an attack classification.
- The attacked data or intruded data is discarded and then the non-intruded data is stored. This storage is done using encryption algorithms to provide secured storage. An optimized AES security algorithm is provided to maintain security.
- The proposed model includes optimal key generation, using the LM-GA approach for both encryption and decryption processes. Finally, the results are verified and validated via with and without optimization testing.

The rest of the paper is arranged in a fashion as given below. Section II reviews the literature concerning various IDS using different approaches. Section III deliberates the proposed architecture and its objectives. Further, Section IV explains the hybridML and the novel IDS. Section V describes the security architecture and the optimization concepts. The simulation results and the achievements are discussed in Section VI. Eventually, Section VII concludes the paper.

II. LITERATURE REVIEW

Related Works

In 2021 [24] developed an AES algorithm to secure data named polymorphic variant of AES (P-AES). The AES parameters' values were modified into each novel key. To accomplish this, key-dependent AES parameters were created such as SubBytes, ShiftRows, and MixColumns. The recipient used the encryption key to obtain the specifics of the activities. As a result, polymorphism was realized, and interoperability was preserved. P-AES was effortlessly constructed using existing AES modules, and its performance was nearly identical to that of AES.

In 2021 [25] have introduced an IDS using hybrid semantic DL approaches (HSDL)with the combination of LSTM, CNN, and Support Vector Machine (SVM). To improve cloud storage security, the regular text is encrypted using the AES approach, and the best key of the AES method is chosen via the Crossover-Based Mine Blast (CMBA) optimization algorithm. Using NSL-KDD and UNSW-NB12 datasets, the suggested model achieved accuracy 99.98% and 98.47% respectively.

In 2022 [26] have developed a novel IDS using DL approaches such as Deep Belief Network (DBN) and Particle Swarm Optimization (PSO). Here, the suggested method was verified with various attacks and attained 96.5 percent accuracy. DARPA 1999 was the dataset used and outperformed the other algorithms like Adaptive-Network-Based Fuzzy Inference System (ANFIS), Harris Hawks Optimization (HHO), Fuzzy Genetic Network Programming (GNP).

In 2021 [27] have proposed a network-based IDS to secure the cloud data through ensemble-based ML methods including Boosted tree, bagged tree, subspace discriminant, and RUSBooted by a voting approach to achieve forecasting results. CICIDS 2017 data was used for simulation in CloudSim. From the experimentation, it was obvious that the implemented model attained a considerable detection rate, minimized false alarm rate, and 97.24 percent of accuracy.

In 2018 [28] have suggested a hybrid IDS via Bayesian, Associative, and Decision tree approaches to protect the data from intrusion. The experimentation involved sensors from IDS on each Cloud host computer. In order to identify dispersed threats, these sensors correlate intrusive signals from each Cloud region. The experiments in real-time and offline simulation functionally validated the system.

In 2019 [7] have established a cloud data security model named Heroku.It used AES to encrypt the data and to enhance the strong protection from the third party. Moreover, a dual cloud platform was used to verify the model. Data was more efficient when it came to uploading and downloading tasks in the cloud if a cloud was operating. In addition, the computing delay between information and encryption suggested that there was more data and that the encryption process took longer.

Review

The literature, it is obvious that the previous ML/DL approaches revealed some advantages and complications. Here are the features and its challenges in the existing research and the need for establishing a novel IDS to secure data in the cloud. The P-AES model [23] exposed better accuracy and enhanced avalanche results but it has high computational time and complicated expressions. The HSDL model [24] attained high accuracy and robustness. Yet, it used multiple ML/DL

approaches which were quite time-consuming. The DBN and PSO model [26] accomplished better accuracy and outperformed the existing models. However, it exposed relatively low performance for some attacks and has time-lag issues. The ensemble-based ML method [26] enhanced accuracy and the minimum error still, computation complexity due to multiple approaches limits the performance. The Bayesian, Associative, and Decision tree approaches [28] obtained considerable performance and robustness while for real-time experimentation, it expressed moderate performance. The Heroku with AES model assured better data protection yet, there is no surety about the accuracy of data security. Thereby, there exists a strong reason to develop a novel IDS to ensure cloud data security with the aid of ML/DL approaches.

III. A NOVEL DL-BASED IDS MODEL TO SECURE CLOUD DATA STORAGE

Proposed Architecture

The **Fig 1.** picturizes the overall schematic representation of novel IDS with DL and AES approaches using the proposed LM-GA model. At first, the attack classification is done on the input intrusion data. The classification is done with the utilization of a hybrid CNN-LSTM model and the input text data is pre-processed and balanced to avoid redundancy and vague data using word2vec representation. The pre-processed data is then subjected to the hybrid CNN-LSTM model to get the IDS output. A hybridization of LSTM is employed for training since a high error rate in the CNN is caused by large numbers of packets. resulting in a high classification rate and a low error rate. The IDS monitors the arrival of new data for intrusions, attacks, and other problems before responding with an attack classification. The attacked data or intruded data is discarded and then the non-intruded data is stored. This storage is done by the use of encryption algorithms to provide secured storage. An optimized AES security algorithm is provided to maintain security. The proposed model includes optimal key generation, using the LM-GA approach for both encryption and decryption processes. Also, the cipher text is further secured via the steganography approach. NSL-KDD and UNSW-NB15 are the datasets used to verify the performance of the proposed LM-GA-based IDS by means of average intrusion detection rate, accuracy, precision, recall, and F-Score.

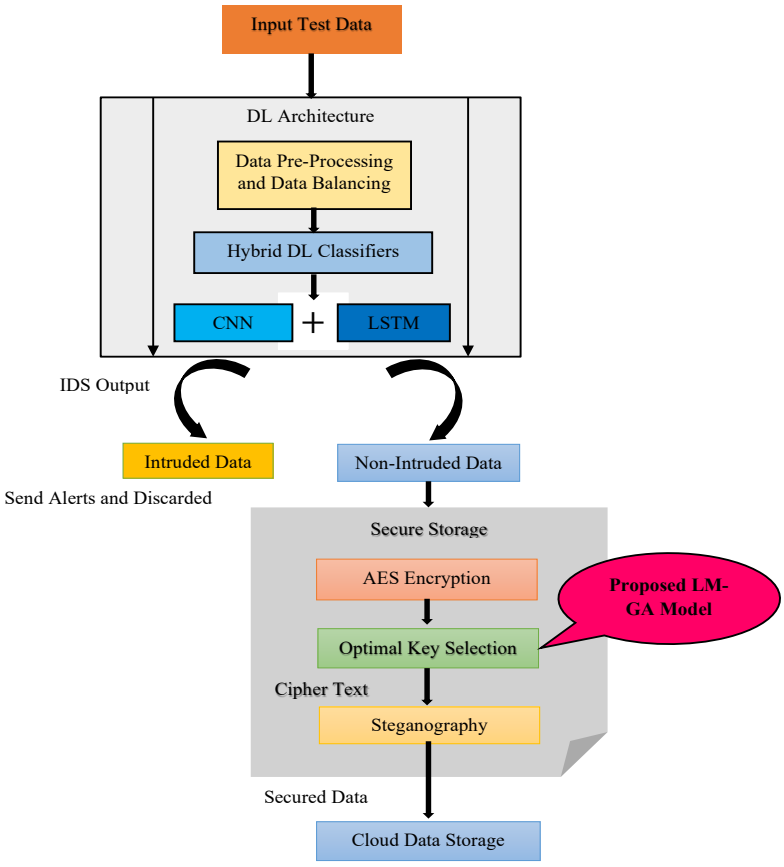

Fig 1. Block Diagram of Novel IDS with DL and AES algorithm using Proposed LM-GA Model

Objective Function

In the process of developing an enhanced cloud storage security system, the significance of AES is improved by tuning it with the efficiency of the proposed LM-GA model. The key objective is to attain the optimal key of AES. Subsequently, the objective is accomplished by scheming the fitness function as given in Eq (1), where Fit is the fitness function of proposed LM-GA, s is each state in AES, and i indicates the number of iterations.

$$Fit = \max\left(key_{breakingtime}\right)_i^s \quad (1)$$

Solution Encoding

Since finding the optimal key is the objective, the initial solution to the proposed LM-GA model is designed as shown in **Fig 2.** Here, $K_1, K_2, \ldots, K_n$ specifies the keys and K portrays the optimal key, and n is the overall key count.

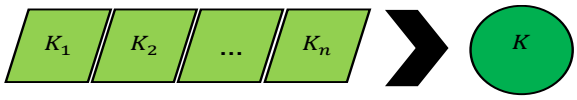

Fig 2. Solution Encoding of Proposed LM-GA Model

Hybrid DL Classifiers and the IDS Concepts

Word2Vec Representation

The enter textual content records are pre-processed and balanced to keep away from redundancy and vague data using word2vec representation. Word2Vec is a hard and fast model that can be used to determine the relationship between a word and the words around it. It's also used to figure out the way to analyze the embeddings of word in a large dataset. The Word2Vec algorithm is used for Natural Language Processing (NLP) situations. The embeddings of words are the textual word of the word vectors. The entered data is the words discovered within the incursion textual content, and the output is the similar word vector.

Traditional CNN Architecture

The word vector is now given as input to the CNN classifier. Generally, CNN [29] is an efficient ML algorithm that comes under the DL category. The fully connected layer of CNN is the final layer that is responsible for finalizing the classification process. Eq. (2) gives the mathematical model of CNN on determining the convolutional feature$A_{x,y,z}^m$ maps with respect to the feature values, where (x, y) indicates the position of the feature values at z^{th} feature map in m^{th}layer. In addition, $W_z^{m\tau}$, $B_{x,y}^m$, and c_z^mspecifies the weight vector, bias factor, and input patch respectively.

$$A_{x,y,z}^m = W_z^{m\tau} B_{x,y}^m + c_z^m \quad (2)$$

The activation function $f_{x,y,z}^m$ is calculated based on convolutional feature $A_{x,y,z}^m$ of CNN as stated in Eq (3)

$$f_{x,y,z}^m = f\left(A_{x,y,z}^m\right) \quad (3)$$

Besides, the pooling function $P_{x,y,z}^m$of CNN is indicated as pool$(\cdot)$ for all feature maps $f_{:,:,z}^m$ and is defined in Eq (4), where $S_{x,y}$ refers to the local neighborhoodatposition(x, y).

$$P_{x,y,z}^m = pool\left(f_{a,b,z}^m\right), \forall (a,b) \in S_{x,y} \quad (4)$$

The loss function $\mathbb{L}$ of CNN can be determined as portrayed in Eq (5), where θ points out each parameter of CNN including the weights $W_z^{m\tau}$, and bias $B_{x,y}^m$. Moreover, n represents the number of chosen input-output relations where $\left\{\left(u^{(i)}, v^{(i)}; i \in [1,2, \ldots n]\right)\right\}, u^{(i)}$ denotes the i^{th}input data, $v^{(i)}$ means to equivalent target label, and $h^{(i)}$ indicates the overall output.

$$\mathbb{L} = \frac{1}{n}\sum_{i-1}^{n} \ell\left(\theta; v^{(i)}, h^{(i)}\right) \quad (5)$$

The features $A_{x,y,z}^m$extracted using CNN.

Traditional LSTM Architecture

The word vector is now given as input to the LSTM classifier as same as the CNN classifier. Traditional LSTM [30] is generally a subset of Recurrent NN (RNN) and is a special type of deep NN. The mathematical model of RNN is clearly explained from Eq (6) to Eq (9), where Tillustrates the time, $k$points out the input dataseries, $g$indicates the hidden neuronsin the network, $e$refers to the neuron's output vector. Furthermore, input to the hidden layer that matrix represent Y, hidden to the output layer that matrix denotes Z, hidden layers at different periods of matrix specifies X and $U_T$means to the chance output of the expected fee after normalization.

$$\omega = d + Xg_{T-1} + Y_{k_T} \tag{6}$$

$$g_T = \tan(\omega_T) \tag{7}$$
$$e_T = h + Zl_T \tag{8}$$

$$U_T = \text{softmax}(e_T) \tag{9}$$

Eq (10) to Eq (13) defines the basic operations involved with the three different gates and the memory cells of the LSTM model.

$$I_t = \sigma(W_i \cdot [g_t - 1], k_T + d_i) \tag{10}$$

$$M_t = \tanh(W_m \cdot [g_t - 1], k_T + d_m) \tag{11}$$

$$F_t = \sigma\big(W_f \cdot [g_t - 1], k_T + d_f\big) \tag{12}$$

$$O_t = \sigma(W_o \cdot [g_t - 1], k_T + d_o) \tag{13}$$

Here, the input gate I_t, memory cellsM_t, forget gate F_t, and output gate O_t are determined using the parameters d, W, k_T, and Twhich is the bias vector, weight matrix, input vector, and time respectively. In addition, σ specifies the sigmoid function. Now,U_T is obtained from the LSTM classifier.

Hybrid CNN-LSTM Classifier

Generally, the ML classifiers CNN and LSTM both exhibit efficient performance in classification. However, some limitations like loss of gradient, imprecise classification, and overfitting restrict the performance. In order to overcome the issues, a novel hybridization of CNN and LSTM models is introduced. Herein, the architecture hybridizes the competence of CNN and LSTM to form a deep hybrid algorithm that adopts the ability to handle the text data efficiently. The mathematical model of the hybrid CNN-LSTM is given below. The inputs $T_x_i$and output T_y_i from the hybrid model is defined in Eq. (14) and Eq. (15) respectively.

$$T\_x_i = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1r} \\ \alpha_{21} & \alpha_{22} & \ldots & \alpha_{2r} \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_{s1} & \alpha_{s2} & \ldots & \alpha_{sr} \end{bmatrix} \tag{14}$$

$$T\_y_i = \left[V_{t-p+3}, V_{t-p+4}, \ldots, V_{t+1}, V_{t+2}, \ldots, V_{t+r}\right]^T \tag{15}$$

Now, s specifies the time step and $r$indicates the data features which is the combination of features from $A^m_{x,y,z}$ (CNN) andU_T (LSTM). The k^{th}convolution kernel $C_k$is stated in Eq (16), where a, and b are the number of filters and strides in the filter size respectively.

$$C_k = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1b} \\ c_{21} & c_{22} & \ldots & c_{2b} \\ \cdots & \cdots & \cdots & \cdots \\ c_{a1} & c_{a2} & \ldots & c_{ab} \end{bmatrix} \tag{16}$$

The process among inputs T_x_i and k^{th}convolution kernel C_k is defined in Eq. (17), which refers to the convolutional layer operation, f_{m_i} is the feature map

$$C_k \odot T_{x_i} = f_{m_i} = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1j} \\ \alpha_{21} & \alpha_{22} & \ldots & \alpha_{2j} \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_{i1} & \alpha_{i2} & \ldots & \alpha_{ij} \end{bmatrix} \tag{17}$$

Eq. (18) portrays the components α in the feature map which is attained by multiplying the convolution kernel $C_k$with the receptive field.

$$T\_x_i\_Field = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1b} \\ \alpha_{21} & \alpha_{22} & \ldots & \alpha_{2b} \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_{a1} & \alpha_{a2} & \ldots & \alpha_{ab} \end{bmatrix} \tag{18}$$

The activation function of ReLU is determined based on Eq (19).

$$ReLU_f = \begin{cases} \alpha & \alpha > 0 \\ 0 & \alpha \leq 0 \end{cases} \tag{19}$$

In the maximum pooling layer, the data samples are compressed and expressed as stated in Eq (20), in which $i' = \frac{i}{a'}$, and $j' = \frac{j}{b'}$.

$$P\_x_i = f_{pool}(f_{m_i}) = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1i'} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2j'} \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_{i'1} & \alpha_{i'2} & \cdots & \alpha_{i'j'} \end{bmatrix} \tag{20}$$

In addition, the LSTM layer is comprised of input, output, and forgot gates and memory cells. The forgot gate in LSTM is defined as shown in Eq (21).

$$F_t = \sigma\left(W_f \cdot [g_{t-l}, k_T] + d_f\right) \tag{21}$$

Eq (22) represents the input gate estimation of LSTM. Similarly, Eq (23) and Eq (24) describe the present memory cell, and output gate respectively.

$$I_t = \sigma(W_i \cdot [g_{t-1}, k_T] + d_i) \tag{22}$$

$$M_t = \tanh(W_m \cdot [g_{t-1}, k_T] + d_m) \tag{23}$$

$$O_t = \sigma(W_o \cdot [g_{t-1}, k_T] + d_o) \tag{24}$$

The result of LSTM is determined using the decision of O_t and $M_t$as given in Eq (25).

$$Z = O_t \circ tanh\, M_t \tag{25}$$

After performing a normalization, the training samples $T_x_i$and T_y_i are subject to the developed CNN model for training the significant parameters with loss function $\mathbb{L}$ and the optimizer ("Adam" is the optimizer used here). The feature map $f_{m_i}$is accomplished and reshaped to train the LSTM. Finally, the classified output Z is attained and the intruded data are discarded and sent a warning about the intrusion.

IV. SECURITY ARCHITECTURE

Traditional AES Algorithm

Now, the non-intruded data is further subjected to secure the data using the AES model. Usually, AES [7] cannot employ a Feistel network, dissimilar to its predecessor DES. The block and key sizes of Rijndael variations are multiple for 32 bits with maximum of 256 bits and a minimum of 128 bits . Usually it uses variation with block size 128-bit, 128, 192 0r 256 bits of key size.

Encryption and Decryption Process: Four phases of permutation and substitution functions are implemented to perform the encryption and decryption process in AES.

SubBytes: During this stage, 256-byte lookup table of S-box is replaced a byte to every byte in the state.

ShiftRows: The state's four rows of bytes have been cylindrically moved to the $(b-1)$ position, in which $b$refers to the number of rows between$1\ to\ 4$.

MixColumn: A Galois Field (GF) multiplication is used to multiply 4×4 matrix various columns and a GF(28) multiplication is employed to attain column-by-column multiplication.

AddRoundkey: An X-OR operation performs with add round key and the present state to the state. The one round key is produced via the preceding sub key used for processing. An encryption method is used to select the key for every round.

AES Key Expansion: The encryption key is enlarged to create a distinct key for each round. Similar to the state, the encryption key is represented as a 4×4 matrix. A 4-byte word makes up each column in the matrix. The SubWord and RotWord transformations are used to enlarge the key. Before the XOR operation, the SubWord and RotWord transformations are applied. Despite the fact that a high level of security provides by AES, security of that algorithm is primarily ensured when it is used with the right key management system. For this purpose, a novel LM-GA method is used to construct an optimum key management system to solve this challenge.

Traditional LA

LA [31] is developed based on the biological life behavior of lions. Here, the male lion is K_{male}, a female lion is K_{female}, and nomad lion is K_{nomad}.There are two main processes namely crossover and mutation and one auxiliary process such as gender clustering is concerned in the mating process. Hither, K_{male}, and K_{female} produce up to four cubs after computing crossover. Furthermore, cubs are the solutions attained from both K_{male}, and K_{female}. Every cub is produced through a different crossover mask C_m. In other words, m^{th} mask C_m is utilized for producing $K^{cubs}(m)$. Further, these

cubs K^{cubs} are subjected to mutation, and produce another four cubs namely K^{new}. Subsequently, these all cubs are placed in the cubs' pool and the gender clustering process takes place to decide K^{m_cubs} and K^{f_cubs}. Now the population (keys) K are generated using LA.

Traditional GA

Similar to all other optimization concepts, GA [32] begins with population initialization, fitness estimation, and fitness. The termination of GA is carried out by convergence like other optimizations as well. Nevertheless, the operations in between the GA vary with other optimization concepts. Generally, GA is comprised of S chromosomes with population R, and for each chromosome, fitness is estimated. Followed with fitness, the selection for mates takes place to perform crossover and mutation. These are the main three steps in GA. The mathematical model of GA is expressed in Eq (26), where $P(a_i)$ is the probability of selected individual S.

$$P(a_i) = \frac{f(a_i)}{\sum_{j=1}^{n} f(a_i)} \tag{26}$$

In many cases, the $P(a_i)$ chosen as unmodified from two-parent chromosomes N and produces no new solutions. Moreover, the genes of the child modify arbitrarily, and mutation is normally slow in GA.

Proposed LM-GA

Generally, both GA and LA expose efficient performance in solving complex problems. However, GA performance is limited by a slow mutation process and random selection of genes for the child might produce an insignificant child. On the other hand, LA is time-consuming and can lead to expensive computation. In order to solve these limitations, this paper intends to introduce a hybridization of GA and LA models to achieve the desired output. For this reason, the slow crossover and mutation process of GA is replaced by the efficiency of LA. That is, the child generation of GA from S chromosomes is replaced with cub generation of LA with respect to K. **Algorithm 1** shows the pseudocode of the proposed model.

Algorithm 1. Pseudocode of Proposed LM-GA Model

Begin

Population Initialization, K $population$, and S $chromosomes$

Assign the parameters like crossover rate C_r, mutation rate m_r, and maximum iterations m_{it}

Fitness estimation

Compute population $P(K)$, selection-reproduction according to GA, C_r, and m_r with respect to LA

Fitness estimation, else perform parameter assignment

Choose the optimal individual

$$iter = iter + 1$$

End

Here, the best key having high key breaking time has been attained and the encrypted data is further protected via the steganography approach.

Steganography Technique

Steganography is a technology that hides data within data to give an extra degree of protection to encryption. The steganography approach uses encrypted sensitive data as input. The sensitive data is concealed through the steganography approach, which employs the best key created in conjunction with the encrypted data to be hidden. This process produces stego-data, which is subsequently saved in a distant cloud database.

V. SIMULATION RESULTS

Simulation Setup

The novel IDS using the proposed LM-GA approach was implemented in MATLAB R2020a on Intel core® core i3 processor 7020U@2.3 GHz, 8 GB RAM, 64-bit operating system. Here, NSL-KDD and UNSW-NB15 Datasets were employed for intrusion classification. The efficiency and novelty of the implemented model were recorded via the simulation results. Here, the evaluation was implemented through various performance parameters such as accuracy, precision, F_1-score, recall, average intrusion detection rate, encryption, and decryption time for both datasets. Furthermore, the efficacy of the implemented approach is compared using with and without optimization concepts.

Algorithmic Analysis

The proposed IDS with novel LM-GA attains enhanced performance by means of average intrusion detection rate for both NSL-KDD and UNSW-NB15datasets. **Table 1** summarizes the average intrusion detection rate for NSL-KDD and UNSW-NB15datasets. For the NSL-KDD dataset, the IDS using hybrid CNN-LSTM-AES approach attained 96.99% which is 7.06%, 7.36%, and 5.28% enhanced than CNN-AES, Two-Stage DL (TSDL) technique [8], and LSTM-AES

respectively. Similarly, the hybrid CNN-LSTM-AES approach accomplished 4.1% better than CNN-AES, 6.8% better than TSDL, and 3.01% improved than LSTM-AES for the UNSW-NB15 dataset.

Table 1. Average Intrusion Detection Rate for NSL-KDD and UNSW-NB15Datasets

Methods	Average Intrusion Detection Rate (%)	
	NSL-KDD	UNSW-NB15
CNN-AES	90.14	91.54
TSDL [8]	89.85	88.99
LSTM-AES	91.86	92.57
CNN-LSTM-AES	96.99	95.45

Fig 3. shows the performance analysis of novel IDS with hybrid CNN-LSTM model with CNN, LSTM, and TSDL using NSL-KDD and UNSW-NB15 datasets by means of accuracy, precision, recall, and F_1-score. The accuracy performance of the hybrid CNN-LSTM model attained 3.26%, 4.99, and 5.02% better than CNN, LSTM, and TSDL respectively. Furthermore, the precision of hybrid CNN-LSTM accomplished 4.15% better than CNN, 5.24% improved than LSTM, and 5.55% improved than TSDL for the NSL-KDD dataset. On the other process, the UNSW-NB15 dataset using the hybrid CNN-LSTM model attained 4.45%, 5.12%, and 5.55% better than CNN, LSTM, and TSDL respectively by means of accuracy. For recall, the hybrid CNN-LSTM model gets 4.75% better than CNN, 5.74% better than LSTM, and 5.95% improved than TSDL. Thereby, the performance analysis verified and validated the efficiency of the hybrid CNN-LSTM model and outperformed the other models.

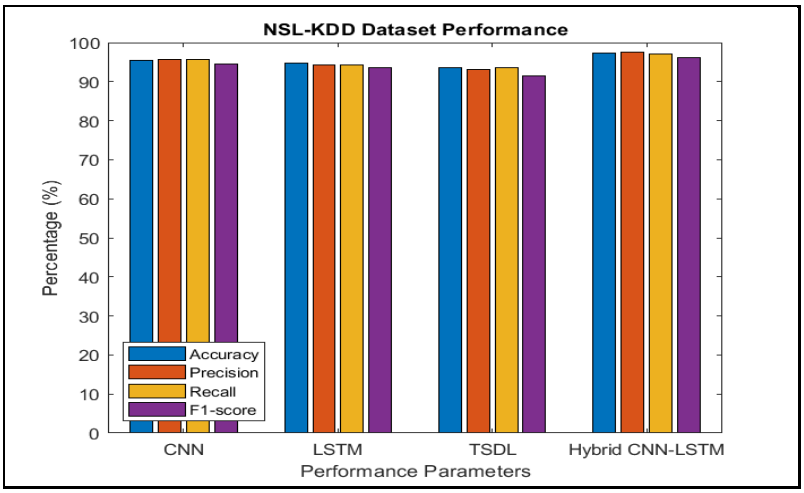

Fig 3(a).

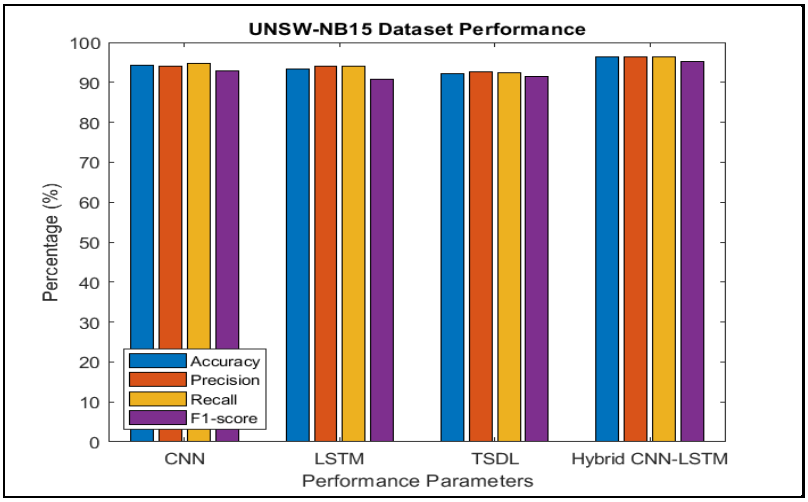

Fig 3(b).

Fig 3. Performance Analysis of IDS with Hybrid CNN-LSTM Modelwith CNN, LSTM, and TSDL in terms of Accuracy, Precision, Recall, and F_1-score for **Fig 3(a).** NSL-KDD and **Fig 3(b).** UNSW-NB15Datasets

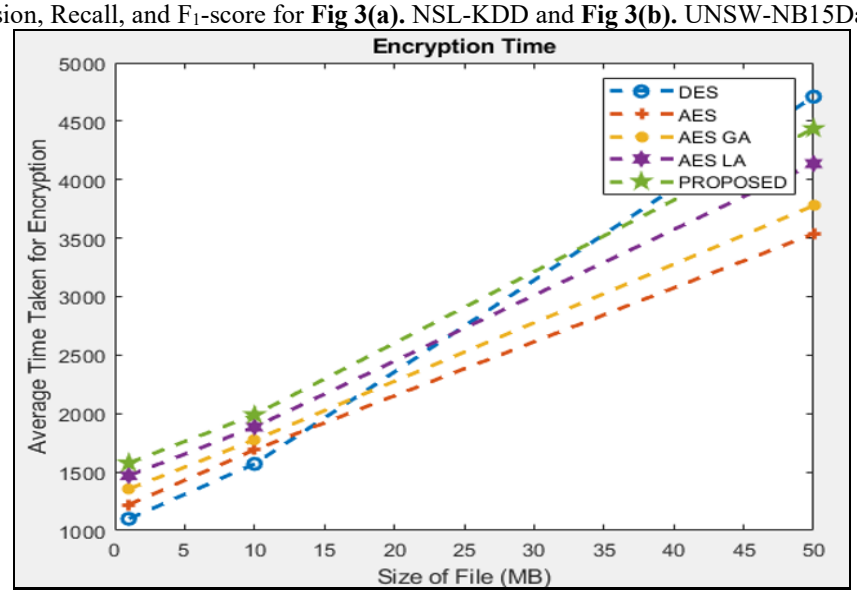

Fig 4(a).

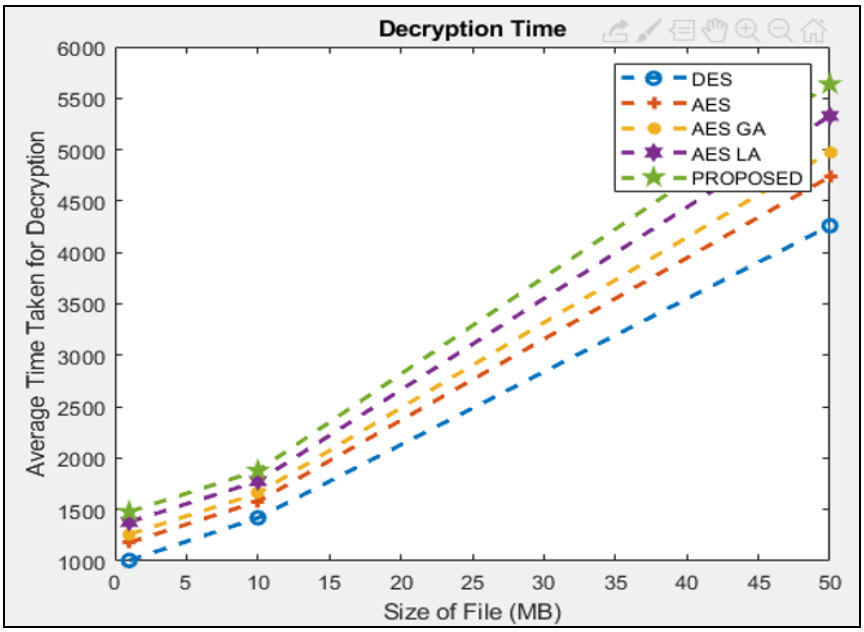

Fig 4(b).

Fig 4. Comparative Analysis for Average Time taken of Proposed IDS with CNN-LSTM-AES-LM-GA Model using with and without Optimization Concepts in Encryption and Decryption for **Fig 4(a).** NSL-KDD and **Fig 4(b).** UNSW-NB15 Datasets

Fig 4. demonstrates the comparative analysis for means of average time taken of proposed IDS with CNN-LSTM-AES-LM-GA Model using with and without optimization concepts for encryption and decryption for NSL-KDD and UNSW-NB15 datasets. The average time taken for encryption of NSL-KDD dataset using CNN-LSTM-AES-LM-GA model accomplished enhanced performance when compared with other methods such as Data Encryption Standard (DES) [35], AES, AES-GA, and AES-LA [38]. Likewise, the average decryption time of the UNSW-NB15 dataset gets better performance using the CNN-LSTM-AES-LM-GA model when compared with other methods like DES, AES, AES-GA, and AES-LA. Finally, **Fig 5.** delivers the converge graph of the proposed LM-GA model with respect to maximum key breaking time [36]. Here, the proposed LM-GA model attained higher key breaking time when compared to other optimization algorithms like Gray Wolf Optimization (GWO) [33], PSO [34], GA, and LA. For iteration 60, the proposed LM-GA model achieved a higher key breaking time which is 4.61% better than GWO, 5.66% better than PSO, 3.91%

improved than GA, and 3.11% better than LA [37]. Thus, the proposed LM-GA shows competence and outperformed the existing optimization algorithms.

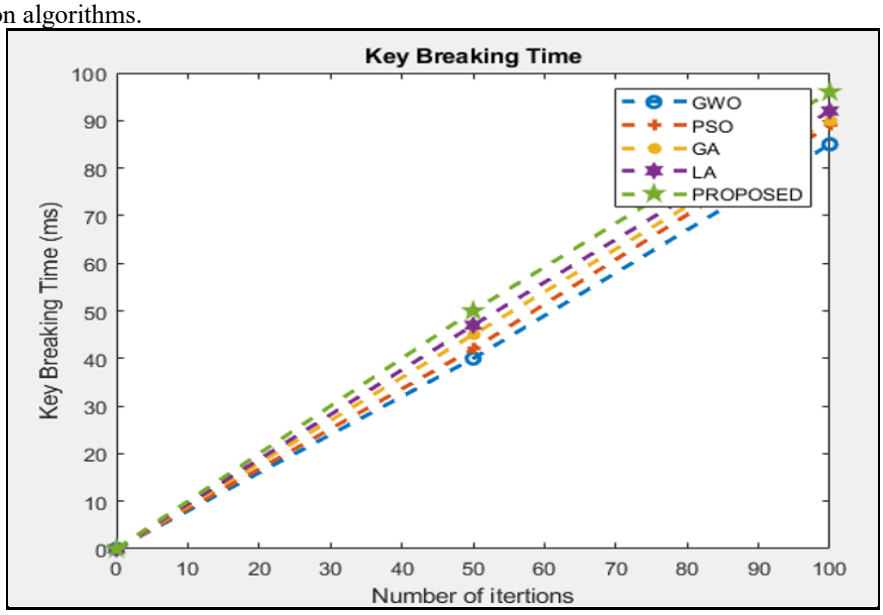

Fig 5. Convergence Graph of Proposed LM-GA Model over Other Optimization Algorithms like GWO, PSO, GA, and LA

VI. CONCLUSION

This paper introduced a novel LM-GA model to attain optimal keys for the AES algorithm. Moreover, an efficient IDS has been established with the efficiency of the hybrid DL architecture namely the CNN-LSTM model. The vague data were processed and converted to a word to a vector representation using the word2vec approach. The processed data was given as input to the CNN and LSTM classifiers and attained the intrusion classified results. The intruded data were discarded, and the non-intruded data were further subjected to AES algorithm to secure the data using optimal key selection process done by the proposed LM-GA model. Finally, remote cloud storage was used to store data after that security ensured by steganography techniques. The efficacy of the implemented IDS was confirmed through NSL-KDD and UNSW-NB15 datasets and attained 96.45% and 95.45% average intrusion detection rates for both datasets. Besides, the accuracy performance of novel IDS accomplished 97.23% and 96.99% for NSL-KDD and UNSW-NB15 datasets respectively. Thereby, the proposed IDS with the hybrid CNN-LSTM-AES-LM-GA model achieved better performance and outperformed the other optimization concepts.

Data Availability Statement

There is no data associated with this article.

Conflict of Interest

The authors declare that there is no conflict of interest.

Funding

No funding was received to assist with the preparation of this manuscript.

Ethics Approval and Consent to Participate

The research has consent for Ethical Approval and Consent to participate.

Competing Interests

There are no competing interests.

References

[1]. S. Reddy, S. c. Krishna, A. Mahendra and N. Neelima, "Providing Security to private cloud Using AES Algorithm," Third International Conference on Inventive Research in Computing Applications (ICIRCA), 2021, pp. 1-5.

[2]. R. U. Khan X. Zhang M. Alazab and R. Kumar "An improved convolutional neural network model for intrusion detection in networks". Proc. Cybersecurity Cyberforensics Conf. (CCC), May 2019, pp. 74-77.

[3]. S. Soheily-Khah P. Marteau and N. Béchet "Intrusion detection in network systems through hybrid supervised and unsupervised machine learning process: A case study on the ISCX dataset", Proc. 1st Int. Conf. Data Intell. Secur. (ICDIS), Apr 2018, pp. 219-226.

[4]. N. Suganya, R. Sathiya, G. Ilamurugan, M. Pavithra and C. Karthikeyan, "Enhancing the Reliability of Cloud Data by Implementing AES Algorithm", 5th International Conference on Intelligent Computing and Control Systems (ICICCS), 2021, pp. 90-95.

[5]. K. Kalaiselvi and A. Kumar, "Enhanced AES cryptosystem by using genetic algorithm and neural network in S-box", IEEE International Conference on Current Trends in Advanced Computing (ICCTAC), 2016, pp. 1-6.

[6]. K. Jaspin, S. Selvan, S. Sahana and G. Thanmai, "Efficient and Secure File Transfer in Cloud Through Double Encryption Using AES and RSA Algorithm," International Conference on Emerging Smart Computing and Informatics (ESCI), 2021.

[7]. P. Sivakumar, M. NandhaKumar, R. Jayaraj and A. S. Kumaran, "Securing Data and Reducing the Time Traffic Using AES Encryption with Dual Cloud", 2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN), 2019, pp. 1-5.

[8]. Daoud L, Hussein F and Rafla N, "Optimization of advanced encryption standard (AES) using vivado high level synthesis (HLS)", In Proceedings of the 34th International Conference on Computers and their Applications, 2019, pp. 36–44.

[9]. Modi, C.N., and Acha, K, "Virtualization layer security challenges and intrusion detection/prevention systems in cloud computing: a comprehensive review", J Supercomput, 2017, pp. 1192–1234.

[10]. O. Alkadi, N. Moustafa, B. Turnbull and K. -K. R. Choo, "A Deep Blockchain Framework-Enabled Collaborative Intrusion Detection for Protecting IoT and Cloud Networks", IEEE Internet of Things Journal, vol. 8, no. 12, 15 June 2021, pp. 9463-9472.

[11]. M. Nadeem, A. Arshad, S. Riaz, S. S. Band and A. Mosavi, "Intercept the Cloud Network From Brute Force and DDoS Attacks via Intrusion Detection and Prevention System," in IEEE Access, vol. 9, 2021, pp. 152300-152309.

[12]. L. Kacha and Abdelhafi Zitouni "An OverView on Data Security in Cloud Computing", Cybern. Approaches Intell Syst. vol. 661, 2017, pp. 250-261.

[13]. Yung Ming and Lily Yuan, "Supervised Learning Methods and Applications in Medical Research", Journal of Computing and Natural Science, vol.2, no.1, pp. 027-034, January 2022. doi: 10.53759/181X/JCNS202202005.

[14]. Y. Gao, Y. Liu, Y. Jin, J. Chen and H. Wu, "A Novel Semi-Supervised Learning Approach for Network Intrusion Detection on Cloud-Based Robotic System", IEEE Access, vol. 6, 2018, pp. 50927-50938.

[15]. G. Loukas, T. Vuong, R. Heartfield, G. Sakellari, Y. Yoon and D. Gan, "Cloud-Based Cyber-Physical Intrusion Detection for Vehicles Using Deep Learning", IEEE Access, vol. 6, 2018, pp. 3491-3508.

[16]. Deshpande, P., Sharma, S.C., Peddoju, S.K, "HIDS: A host based intrusion detection system for cloud computing environment", Int J Syst Assur Eng Manag, vol. 9, 2018, pp. 567–576.

[17]. Jaber, A.N., Rehman, S.U. "FCM–SVM based intrusion detection system for cloud computing environment", Cluster Comput, vol. 23, 2020, pp. 3221–3231.

[18]. Krishnaveni, S., Sivamohan, S., Sridhar, S.S. et al. "Efficient feature selection and classification through ensemble method for network intrusion detection on cloud computing", Cluster Comput, vol. 24, 2021, pp. 1761–1779.

[19]. Wei, P., Li, Y., Zhang, Z., Hu, T., Li, Z., & Liu, D. "An optimization method for intrusion detection classification model based on deep belief network", IEEE Access, vol. 7, 2019, pp. 87593–87605.

[20]. M. Mazini B. Shirazi and I. Mahdavi, "Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms", J. King Saud Univ.-Comput. Inf. Sci, vol. 31, no. 4, October 2019, pp. 541-553.

[21]. P. Narwal D. Kumar and S. N. Singh "A hidden Markov model combined with Markov games for intrusion detection in cloud", J. Cases Inf. Technol, vol. 21, no. 4, October 2019, pp. 14-26.

[22]. Wen, L. "Cloud Computing Intrusion Detection Technology Based on BP-NN", Wireless Pers Commun (2021).

[23]. Z. Chkirbene, A. Erbad, R. Hamila, A. Mohamed, M. Guizani and M. Hamdi, "TIDCS: A Dynamic Intrusion Detection and Classification System Based Feature Selection", IEEE Access, vol. 8, 2020, pp. 95864-95877.

[24]. R. Priyadharshini and E. J. Leavline, "Cuckoo optimisation based intrusion detection system for cloud computing", Int. J. Comput. Netw. Inf. Secur, vol. 10, no. 11, 2018, pp. 42-49 Nov.

[25]. 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, 2021, pp. 20191-20207.

[26]. Prabhakaran, V, Kulandasamy, A. "Hybrid semantic deep learning architecture and optimal advanced encryption standard key management scheme for secure cloud storage and intrusion detection", Neural Comput & Applic vol. 33, 2021, pp. 14459–14479.

[27]. Sajith, P.J., Nagarajan, G. "Intrusion Detection System Using Deep Belief Network & Particle Swarm Optimization", Wireless Pers Commun (2022).

[28]. Arulkumar, V. et al. "A performance enhancement of deepfake video detection through the use of a hybrid CNN Deep learning model", International Journal of Electrical and Computer Engineering Systems. Vol. 14 No. 2 (2023). https://doi.org/10.32985/ijeces.14.2.6.

[29]. Modi, C., Patel, D. "A feasible approach to intrusion detection in virtual network layer of Cloud computing", Sādhanā, vol. 43, 2018, pp. 114-121.

[30]. Wei He, Jufeng Li, Zhihe Tang, Beng Wu, Hui Luan, Chong Chen, and Huaqing Liang, "A Novel Hybrid CNN-LSTM Scheme for Nitrogen Oxide Emission Prediction in FCC Unit", Mathematical Problems in Engineering, Vol. 20, 17 August 2020.

[31]. Aruna, M, "Medical Healthcare System with Hybrid Block based Predictive models for Quality preserving in Medical Images using Machine Learning Techniques." 2022 International Conference on Advanced Computing Technologies and Applications (ICACTA). IEEE, 2022.

[32]. Rajakumar Boothalingam, "Optimization using lion algorithm: a biological inspiration from lion's social behaviour", Evolutionary Intelligence, Vol. 11, 2018, pp. 31–52.

[33]. Yi Ding, and Xian Fu, "Kernel-Based Fuzzy C-Means Clustering Algorithm Based on Genetic Algorithm", Neurocomputing, Vol. 188, 5 May 2016, pp. 233-238.

[34]. Seyedali Mirjalili, Seyed Mohammad Mirjalili and Andrew Lewisa, "Grey Wolf Optimizer", Advances in Engineering Software, Vol. 69, March 2014, pp 46-61.

[35]. Rimma Padovano, "Critical Analysis of Parallel and Distributed Computing and Future Research Direction of Cloud Computing", Journal of Computing and Natural Science, vol.1, no.4, pp. 114-120, October 2021. doi: 10.53759/181X/JCNS202101017.

[36]. Nobile, M.S; Cazzaniga, P.; Besozzi, D.; Colombo, R.; Mauri, G.; Pasi, G., "Fuzzy Self-Tuning PSO: a settings-free algorithm for global optimization", Swarm and Evolutionary Computation, Vol.39, pp. 70–85. 2018

[37]. Ratnadewi, Roy Pramono Adhie, Yonatan Hutama, A. Saleh Ahmar and M I Setiawan, "Implementation Cryptography Data Encryption Standard (DES) and Triple Data Encryption Standard (3DES) Method in Communication System Based Near Field Communication (NFC)", Journal of Physics: Conference Series, Vol. 954.2018.

[38]. F. Thabit, O. Can, S. Alhomdy, G. H. Al-Gaphari, and S. Jagtap, "A Novel Effective Lightweight Homomorphic Cryptographic Algorithm for data security in cloud computing," International Journal of Intelligent Networks, vol. 3, pp. 16–30, 2022, doi: 10.1016/j.ijin.2022.04.001.