THE DEVELOPMENT OF MAN-IN-THE-MIDDLE THREAT DETECTION MODELS WITH NEURAL NETWORKS

[1]OLGA USSATOVA, [2]AIDANA ZHUMABEKOVA, [3]VLADISLAV KARYUKIN, [4]YENLIK BEGIMBAYEVA, [5]NIKITA USSATOV, [6]ERIC T. MATSON

[1,2,3,4,5]Institute of Information and Computational Technologies, Almaty, Kazakhstan
[1]Almaty University of Power Engineering and Telecommunications named after G. Daukeyev, Almaty, Kazakhstan
[2,3]Al-Farabi Kazakh National University, Almaty, Kazakhstan
[4]Satbayev University, Almaty, Kazakhstan
[5]Turan University, Almaty, Kazakhstan
[6]Purdue University, West Lafayette, United States
E-mail: zhumabekova2702@gmail.com

Abstract - In the digital era, the spontaneous growth of Internet technologies has led to the significant importance of the information security field. The complexity of online threats and attacks puts the users' computer infrastructure at great risk. Man-in-the-Middle attacks are among the most common threats to users' data integrity, confidentiality, and availability. Although traditional methods still give opportunities to analyze the network traffic to detect Man-in-the-Middle attacks, they are becoming less relevant with the rapid development of intelligent tools. Therefore, new approaches based on neural networks offer powerful instruments to reveal anomaly traffic and discover hidden patterns in attacks. In this work, Dense, Convolutional, Long short-term memory, and a hybrid model of Convolutional and long short-term memory neural networks are utilized to detect Man-in-the-Middle attacks. The dataset of 2504267 instances (1145272 normal and 1358995 malicious) is used to conduct the experiments. All four types of neural networks showed efficiency with accuracy scores of 0.96 and above, and the CNN-LSTM neural network achieved the best score of 0.998. These experiments demonstrated how a deep learning approach could be beneficial in detecting Man-in-the-Middle attacks. These models are also important in analyzing other kinds of threats and attacks, which will be covered in subsequent research.

Keywords - Internet technologies, Man-in-the-Middle attacks, Dense neural network, Convolutional neural network, Long short-term memory network, CNN-LSTM

I. INTRODUCTION

In the era of advanced digital technologies, the importance of information security is increasing.This is because the number and sophistication of online threats are growing every day, putting big data at risk and becoming a major cybersecurity issue [1].Man-in-the-middle attacks are one of the most common threats to the integrity, confidentiality, and availability of information.The MitM attack type is one of the most secretive and difficult to detect.In this type of threat, an attacker can secretly monitor and control communications between two parties.This attack will likely have serious consequences, including leaking sensitive information about organizations and individuals and compromising system security [2].Due to the complexity and stealth of the MitM attack, prevention and timely detection of the threat is key.

Detecting MitM attacks by analyzing network traffic and using cryptographic protocols is less effective.The growth and variety of information transmitted through networks, as well as the development of methods used by attackers, require the implementation of intelligent solutions.Machine learning methods and neural networks are promising ways to improve attack detection systems' efficiency [3].In this regard, neural networks offer a powerful tool for network traffic analysis and anomaly detection, with the critical ability to adapt to new types of attacks, learn from big data, and discover hidden patterns.

Recently, the use of neural network methods to ensure information security and identify threats has become widespread.These include convolutional neural networks, long short-term memory, deep neural networks, hybrid convolutional models, and many others.Each of these models has its own features and advantages that allow you to detect deviations in network traffic and identify signs of a MitM attack.

II. LITERATURE REVIEW

The security of smart grid applications is explored in [4], where the combination of machine learning anomaly detection with signature-based methods was determined. This approach became especially effective in revealing Man-in-the-middle attacks. The laboratory experiments showed a high detection accuracy rate of 97% to 100%.[5] classified several types of machine learning algorithms on the SSADA pressure dataset, including decision trees, discriminant analysis, logistic regression, Naive Bayes, support vector machines, nearest neighbors, and ensemble learning algorithms. For detecting MitM attacks in the oil and gas process control network, research work conducted on 68,722 datasets concluded that the

CoarseTree method of the decision tree algorithm performed well and was the most suitable. The authors of MiTM and DoS attacks [6] performed classification using four machine learning algorithms on datasets obtained from Kaggle. According to the proposed indicators, XGBoost, Random Forest, Decision Tree, and Gradient Boosting algorithms were comparatively better for MitM attacks than DoS attacks.The paper [7] used machine learning algorithms Random Forest (RF), Extreme Gradient Boosting (XGBoost), Gradient Boosting (GB), and Decision Tree (DT) to detect MiTM attacks according to accuracy, precision, recall, and f1-score metrics detected the MiTM attack with very high, that is, more than 99% identical performance.Protecting military ground robots from MitM attacks in the research work [8] involves the use of deep convolutional neural networks (CNN). Using the proposed method for detecting cyber-attacks in real time, good results were achieved, and the effectiveness of this method was determined. The authors propose this method to increase the reliability of robots' actions on the battlefield. In [9], a Secure Data Aggregation Scheme (SDAS) is proposed to detect the MitM attack and ensure the integrity and accuracy of the collected results, and the authors used the OMNeT++ simulator during the experiment. As a result of the experiment, it was confirmed that using the proposed SDAS, it is possible to effectively identify danger and maintain the reliability of the collected results up to 90%. To detect and prevent the threat in a Mobile Ad-Hoc Network (MANET) susceptible to MitM attacks, the authors of [10] conducted a study using a prediction method in artificial neural networks. The ANN classification method was implemented for attack detection in MANET and used in conjunction with the NS2 simulation platform. The experiments conducted by the authors obtained good results in terms of time and performance of 88.235%.

III. METHODOLOGY

A.Data processing and feature

The initial stage of threat detection involves a significantly important step of gathering the dataset. In this research, the Man in the Middle (MitM) dataset was received from the Kaggle open-source website (https://www.kaggle.com/datasets/ymirsky/network-attack-dataset-kitsune). After the dataset was collected, the next important steps took place, including data preprocessing, cleaning, normalizing, feature selection, and balancing for the following classification with the use of deep learning models. During cleaning, unnecessary data, symbols, and errors were removed, and similar issues were solved.

Then, the data normalization technique, which can mitigate differences between feature values, was applied. This method is beneficial when features have different scales, allowing them to contribute equally to the model's performance. For instance, if two objects have values from 0 to 10 or from 0 to 5000, they can mislead the classification results by making one feature more influential. The widespread Min-Max scaling [11] approach commonly addresses this issue by normalizing features' values. Normalized values are computed as follows (1):

$$X'_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}}, (1)$$

where X is an initial value; X_{min} is a minimum value; X_{max} is a maximum value; X'_{scaled} is a scaled value.

As each dataset has a different number of features, it is necessary to get an optimal number of features for the subsequent steps. After normalization, the feature selection algorithm is applied to get the most significant features. In this work, the twenty best features were chosen for each dataset using the Chi-square feature selection method [12]. This statistical technique is used to select the most relevant feature for classification tasks in machine learning. In the context of feature selection, it evaluates how well each feature can distinguish between different classes. A contingency table for each feature shows the frequency distribution of feature values across various classes. The method is defined by the formula (2):

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}, (2)$$

where O_i is the observed frequency, and E_i is the expected frequency.

E_i is itself calculated as (3):

$$E_i = \frac{(row_total \times column_total)}{grand_total} (3)$$

When the Chi-square values for each feature were computed, they were ordered in descending order, and the twenty best of them were selected.

One essential step after the application of the feature selection approach is balancing. The random oversampling method [13] was utilized to implement this step. This method aims to balance classes by increasing the number of instances in the minority class, making them equal. It is a straightforward method.

Although traditional machine learning approaches are still relevant in the threat classification field, the neural network approach has become dominant in Artificial Intelligence in the last decade. They show high accuracy in speech and image recognition, text

classification, and natural language processing tasks.There are several types of commonly used neural networks: dense neural networks (DNN), convolutional neural networks (CNN), long short-term memory (LSTM), and the combination of convolutional and long short-term memory neural networks called CNN-LSTM.

Dense neural network (DNN)[14] is a neural network model with two or more hidden layers.A neural network consists of an input layer with input data, hidden layers that include node elements called neurons, and an output layer containing one or more neurons.1.

In this model, $x = x_1, x_2, ..., x_f$ is an input vector, $w_1, w_2, ..., w_i$ are the weights of each level, and $b_1, b_2, ..., b_i$ is the bias.Levels from l_2 to l_{n-1} form hidden layers, and the output level l_n is represented by the corresponding output vector $y_1, y_2, ..., y_m$ .The elements of the hidden and output layers are called neurons.They are represented by activation functions responsible for the nonlinear functional mapping between the input data and the response variable.The most popular activation functions are the sigmoid function, hyperbolic tangent function (tanh), rectified linear unit (ReLu), and softmax.The sigmoid function is mainly used on the output layer in binary classification as it defines the output value as 0 or 1. The tanh function is an improved version of the sigmoid function, with the only difference being that in tanh, the output values range from -1 to1. In hidden layers, the ReLu activation function is most often used.This results in an output value of 0 if it receives a negative input x. Otherwise, for positive inputs, it returns x unchanged, like a linear function.Mathematically, this is denoted as $f(x) = \max(0, x)$ .The softmax function is used in multiclass classification by calculating the probability that each occurrence belongs to a predefined class and adjusting the output values for each class to be in the range from 0 to 1. The softmax function is typically used onlyfor the output layer.

Convolutional neural network (CNN)[15] is a class of neural networks primarily used for image classification, image recognition, and other computer vision tasks. Nevertheless, it also gained popularity and is widespread in other fields, particularly cybersecurity threat classification. This type of neural network leverages spatial hierarchies in data using convolutional layers, pooling layers, and fully connected layers.

Generally, the architecture of CNN includes the following elements:

- Convolutional layers are the layers that apply convolutional filters to the input data to create feature maps. This operation involves sliding the filter across the input data and computing the dot product between the filter and input data.
- Max-pooling layers perform operations to reduce feature size by selecting the maximum values of the feature map.
- Fully connected layers present layers that are similar to those in traditional neural networks, where each neuron is connected to neurons of previous layers.

Output layers are the final layers that contain the classification results.In the binary classification problem, the output values are represented by a vector $\vec{y} = y_1, y_2$.

The scheme of the CNN model is shown in Fig. 1.

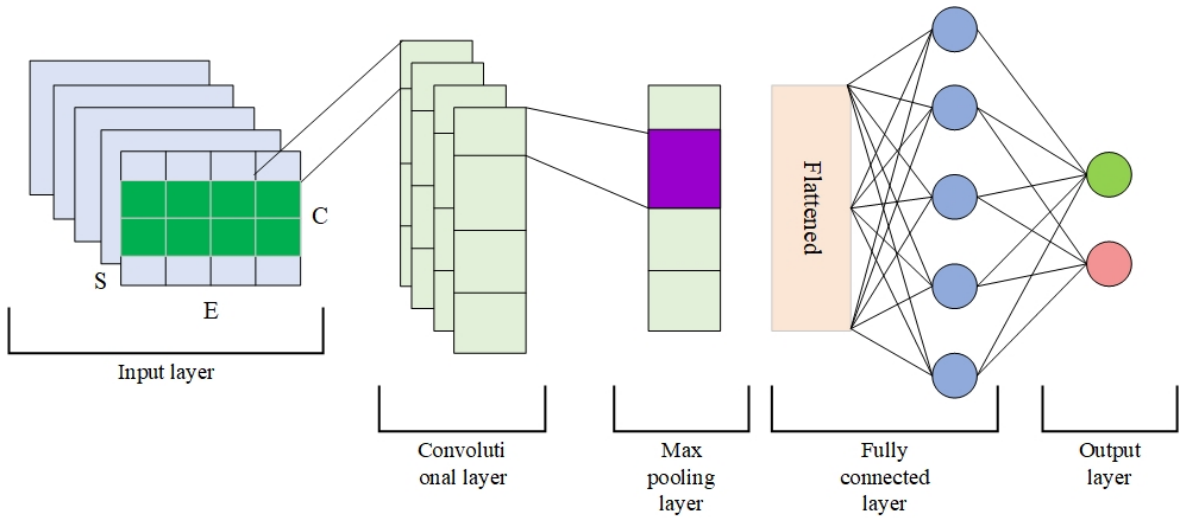


Fig. 1. Convolutional neural network architecture

When the formed matrix $S \times E$ is given, the filter $C \times E$ is applied to the matrix, and the scalar product of the matrix with the filter elements is performed. The resulting value is transferred to the next layer. The max-pooling process is applied to the resulting feature vectors on the next layer to extract the largest value from each vector. After convolutional and pooling layers, the data is processed by fully connected layers. The result of these operations proceeds to the output layer, which is the final layer of the network, where sigmoid or softmax activation functions are applied to get the corresponding results.

Recurrent neural networks (RNN) [16] is a type of neural network designed to process sequential data and consider the context of various sequences.One of the popular types of recurrent neural networks is a neural network with long short-term memory (LSTM).The following elements are presented in the context of LSTM:

The scheme of the RNN model is shown in Fig. 2.

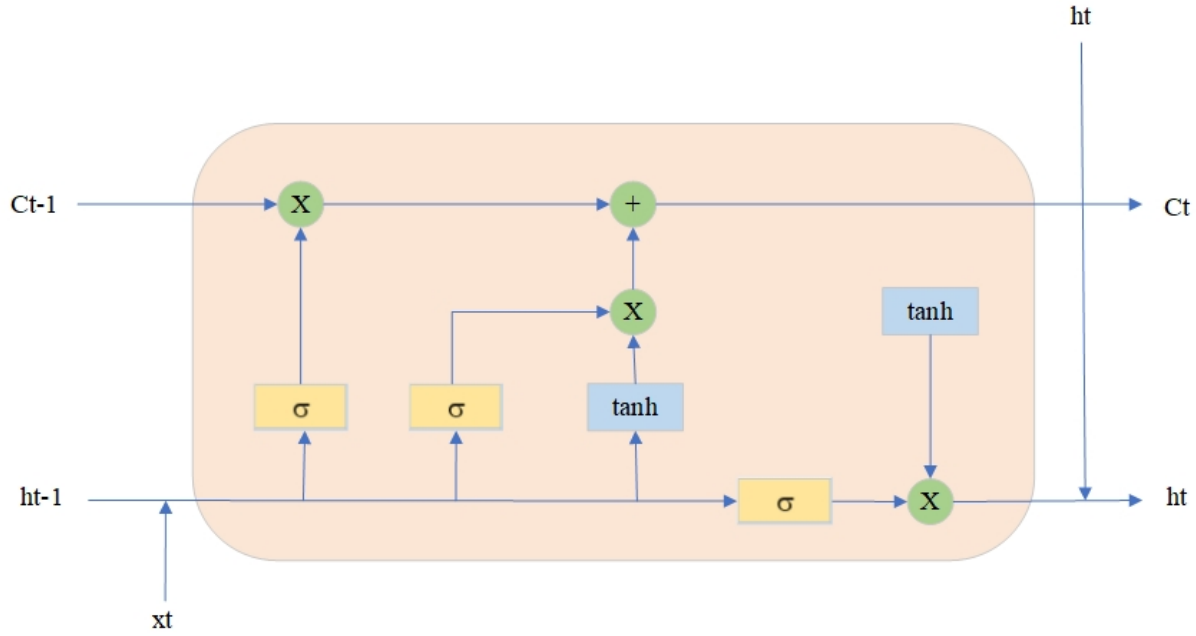


Fig. 2. Recurrent neural network

Convolutional and long short-term memory neural network (CNN-LSTM) [17] is a hybrid combined modelthat includes both CNN and LSTM architectures. This type of neural network supports various data types, including text and numeric data. Its structure consists of an input layer (input data), CNN layers (convolutional and pooling layers that extract spatial features from each time step), Flattening (it converts the output of CNN layers into a suitable shape for LSTM layers), LSTM layers, Dense layers, and Output layers [18].

Each component of this neural network performs important functions:

- A Previous memory cell C_{t-1} is the memory cell at the previous time step.It represents the information that was saved in the previous steps.
- A current memory cell C_t is the state of the current memory cell.It is updated based on the previous memory cell, the input data x_t, and the LSTM gate (Forget Gate, Input Gate) decision that determines what information to keep or forget.
- A previous hidden state h_{t-1} is the hidden state at the previous time step.
- A current hidden state h_t is the current hidden state.
- Previous input data x_{t-1} is input data at the previous time step.
- Current input data x_t is input data at the current time step.

- The input layer presents the input data of the neural network.
- CNN layers extract spatial features from each time step.
- Flattening converts the output of CNN layers into a suitable shape for LSTM layers.
- LSTM layers work with sequential data and can consistently capture long-term dependencies.LSTM has memory and can store information for a long time.
- Dense layers are fully connected layers of neurons that pass data from the previous layer to the next one.
- The output layer presents the output data, which is a class in the classification task.

The CNN-LSTM model is shown in Fig. 3.

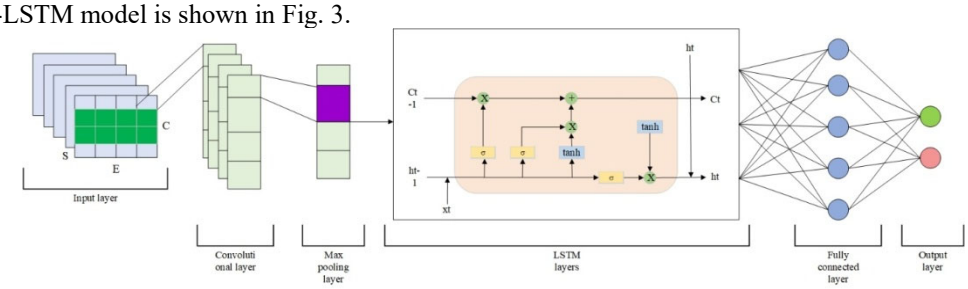


Fig. 3. The combined CNN-LSTM model

IV. EXPERIMENTAL RESULTS

The experimental results of this work focused on the detection of MitM threats [19]. It includes 1145272 normal traffic instances and 1358995 attack traffic instances. This dataset is essential because it contains records of network traffic where attackers intercept and alter the communication between two sides without their knowledge of it.

In the experiments, the first steps included dataset preprocessing, cleaning, normalizing, feature selection, and balancing [20] for the subsequent classification using deep learning models [21]. The classification results were evaluated using accuracy, precision, recall, and F1-score metrics [20]. Accuracy is the simplest metric representing a proportion of correctly classified instances to the total number of instances. It is calculated by (4):

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (4)$$

Precision measures the accuracy of the positive predictions of all instances predicted as positive (5):

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

Recall computes the ability of the model to identify all positive instances, the proportion of true positives out of all actual positive instances (6):

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

F1-score is a harmonic mean of Precision and Recall, balancing values between them.It is calculated by (7):

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (7)$$

Here, TP (True Positives) are correctly predicted positive instances; TN (True Negatives) are correctly predicted negative instances; FP (False Positives) are incorrectly predicted positive instances; FN (False Negatives) are incorrectly predicted negative instances [22].

The experimental results of the classification of MitM threats were evaluated with the mentioned metrics, AUC-ROC (Area Under the Curve and Receiver Operating Characteristic) curves, and histograms [23]. The classification results for the SQL injections are shown in Table 1. Histograms and AUC-ROC curve plots are shown in Fig. 4-5.

Metrics	DNN	CNN	LSTM	CNN-LSTM
Accuracy	0.967	0.973	0.978	0.998
Precision	0.969	0.968	0.977	0.999
Recall	0.965	0.979	0.979	0.997
F1-score	0.967	0.973	0.978	0.998

Table 1:MitM classification metrics

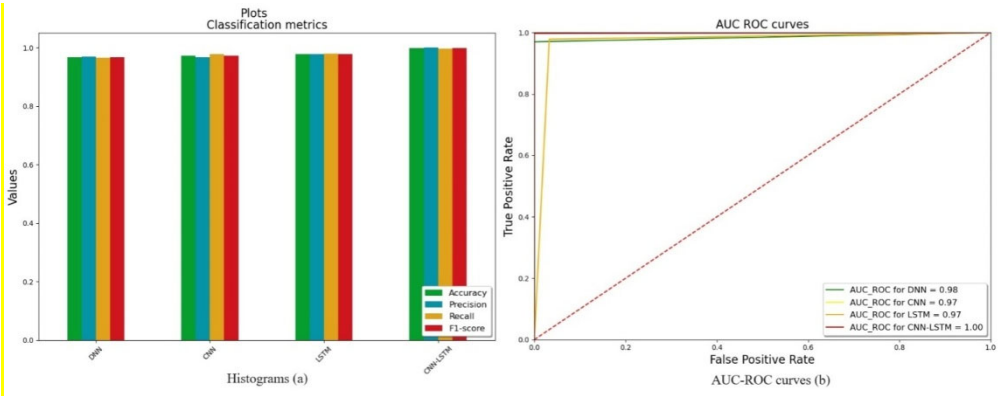


Fig. 4.MitM histograms (a) and AUC-ROC curves (b)

The experimental results showed that all 4 types of deep neural networks allowed to get excellent accuracy scores of 0.96 and above. Among all models, the CNN-LSTM neural network demonstrated the best values of accuracy, precision, recall, and f1-score. Therefore, the deep learning approach proved to be very effective in classifying MitM threats.

V. CONCLUSION

Today, in the digital age, Internet technologies are developing increasingly fast, making the information security field especially significant in protection against various cyber threats and attacks. Among these attacks, MitM attacks are particularly concerning as they jeopardize the integrity, confidentiality, and availability of information. Also, there are various traditional techniques that allow the detection of MitM attacks, but they are becoming less effective, and new advanced approaches based on machine and deep learning methods remain especially meaningful. DNN, CNN, LSTM, and CNN-LSTM models effectively provide adaptive capabilities and mitigate MitM threats. The experimental results demonstrated the stages of training the mentioned 4 types of neural networks in classifying MitM threats. The dataset included 2504267 instances (1145272 normal and 1358995 malicious). All neural networks showed an accuracy score of 0.96 and above, but the CNN-LSTM neural network achieved the best score of 0.998. Future studies are planned to investigate detecting other types of threats, such as DDoS and malware attacks, phishing, and SQL injections.

ACKNOWLEDGMENTS

This research was carried out within the framework of the project AP19675957, “The research and development of the system for ensuring the protection of medical data using blockchain technology and artificial intelligence methods,” which is being implemented at the Institute of Information and Computational Technologies.

REFERENCES

[1] D. Ghelani, “Cyber Security, Cyber Threats, Implications and Future Perspectives: A Review,” American Journal of Science, Engineering and Technology, vol. 3, no. 6, (2022) 12-19.

[2] Muzammil M. B. B. Muhammad, A. Sahar, C.S. Sandile, and Y.G. Yazeed, “Unveiling Vulnerabilities of Web Attacks Considering Man in the Middle Attack and Session Hijacking,” IEEE Access, (2024) 6365-6375.

[3] S. Yuan and X. Wu, “Deep learning for insider threat detection: Review, challenges and opportunities,” Computers & Security, vol. 104 (2021) 102221.

[4] M.F. Elrawy, et al., “Detecting and classifying man-in-the-middle attacks in the private area network of smart grids,” Sustainable Energy, Grids and Networks, vol. 36 (2023) 101167.

[5] U.O. Obonna,F.K. Opara, C.C. Mbaocha, J.K.C Obichere, I.O. Akwukwaegbu, M.M. Amaefule, and C.I. Nwakanma, “Detection of Man-in-the-Middle (MitM) Cyber-Attacks in Oil and Gas Process Control Networks Using Machine Learning Algorithms,” Future Internet, vol. 15, no. 8 (2023) 280.

[6] S.A.M. Al-Juboori, F. Hazzaa, Z.S. Jabbar, S. Salih S., and H. M. Gheni, “Man-in-the-middle and denial of service attacks detection using machine learning algorithms,” Bulletin of Electrical Engineering and Informatics, vol. 12, no. 1 (2023) 418-426.

[7] N. Sakthipriya, V. Govindasamy, and V. Akila, “Security-aware IoT botnet attack detection framework using dilated and cascaded deep learning mechanism with conditional adversarial autoencoder-based features,” Peer-to-Peer Networking and Applications, 17 (2024) 1467–1485.

[8] F. Santoso and A. Finn, “Trusted Operations of a Military Ground Robot in the Face of Man-in-the-Middle Cyber-Attacks Using Deep Learning Convolutional Neural Networks: Real-Time Experimental Outcomes,” IEEE Transactions on Dependable and Secure Computing, vol. 21, no. 4 (2024) 2273-2284.

[9] M.I. Adawy et al., “Man-In-The Middle Attack Detection Scheme on Data Aggregation in Wireless Sensor Networks,” International Journal of Advances in Soft Computing & Its Applications, vol. 15, no. 2 (2023).

[10] R.A. Sowah, et al., “Detection and prevention of man-in-the-middle spoofing attacks in MANETs using predictive techniques in artificial neural networks (ANN),” Journal of Computer Networks and Communications, vol. 2019, no. 1 (2019) 4683982.

[11] E. Ntizikira, L. Wang, J. Chen, and K. Saleem, “Honey-block: Edge assisted ensemble learning model for intrusion detection and prevention using defense mechanism in IoT,” Computer Communications, vol. 214 (2024) 1-17.

[12] G. Rathod, V. Sabnis, and J.K. Jain. Improving IoT Botnet Attack Detection using Machine Learning: Comparative Analysis of Feature Selection Methods and Classifiers in Intrusion Detection Systems, 3rd International Conference for Innovation in Technology (INOCON), Bangalore, India (2024) 1-8.

[13] M.A. Talukder, M.M. Islam, M.A. Uddin. et al. Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction // Journal of Big Data,11 (2024) 33.

[14] M. I. Gulbarga, S. Cankurt, N. Shaidullaev, and A. Khan. Deep Learning (DL) Dense Classifier with Long Short-Term Memory Encoder Detection and Classification against Network Attacks, 2023 17th International Conference on Electronics Computer and Computation (ICECCO), Kaskelen, Kazakhstan (2023) 1-6.

[15] V. Kanimozhi, T.P. Jacob, “The Top Ten Artificial Intelligence-Deep Neural Networks for IoT Intrusion Detection System,” Wireless Personal Communications, 129 (2023) 1451–1470.

[16] C.R. Kishore, D.C. Rao, J. Nayak et al., “Intelligent Intrusion Detection Framework for Anomaly-Based CAN Bus Network Using Bidirectional Long Short-Term Memory,” Journal of the Institution of Engineers (India): Series B, 105 (2024) 541–564.

[17] J. Luo, “Man-in-the-Middle Intrusion Detection Based on CNN-LSTM Model,” IEEE 3rd International Conference on Electronic Technology, Communication and Information (ICETCI), Changchun, China (2023) 1-7.

[18] A. Javed, A. Ehtsham, M. Jawad, M.N. Awais, A.-u.-H. Qureshi, and H. Larijani, “Implementation of Lightweight Machine Learning-Based Intrusion Detection System on IoT Devices of Smart Homes,” Future Internet, 16 (2024) 200.

[19] A. Zhumabekova et al., “Determining Web Application Vulnerabilities Using Machine Learning Methods,”2023 19th International Asian School-Seminar on Optimization Problems of Complex Systems (OPCS), Novosibirsk, Moscow, Russian Federation, (2023) 136-139.

[20] H. Nandanwar and R. Katarya, "Deep learning enabled intrusion detection system for Industrial IOT environment," Expert Systems with Applications, vol. 249 (2024).

[21] O. Ussatova, V. Karyukin, A. Zhumabekova, Y. Begimbayeva, and N. Ussatov. Designing a vulnerability threat detection scanner with the use of machine learning models. In Proceedings of the 13th International Conference on Advances in Information Technology (IAIT '23). Association for Computing Machinery, New York, NY, USA, Article 16 (2023) 1–8.

[22] C. Hazman, A. Guezzaz, S. Benkirane, and M. Azrour, "Enhanced IDS with Deep Learning for IoT-Based Smart Cities Security," Tsinghua Science and Technology, vol. 29, no. 4 (2024) 929-947.

[23] O. Ussatova, A. Zhumabekova, V. Karyukin, E. Matson, and N. Ussatov. The development of a model for the threat detection system with the use of machine learning and neural network methods. International Journal of Innovative Research and Scientific Studies, 7(3) (2024)863–877.

★★★