A Deep Learning-Based Framework for Arp Spoofing Attack Detection

Pavithraa S*[1] and Khanaa V[2]

[1]Department of Computer Science and Engineering, Bharath Institute of higher education and research, Chennai, India.
[2]Department of Information Technology, Bharath Institute of higher education and research, Chennai, India.
pavithraa.it@bharathuniv.ac.in

Abstract. ARP spoofing attacks are a serious risk to network security because they allow malevolent actors to intercept and alter network traffic, which frequently results in data breaches and information leaks. This paper introduces a deep learning-based method for identifying ARP spoofing that makes use of Long ShortTerm Memory (LSTM) networks and Convolutional Neural Networks (CNNs). The ARP traffic dataset was used to train and assess both models, which capitalized on the advantages of CNNs for spatial feature extraction and LSTMs for temporal sequence modeling. Recall, accuracy, precision, F1-score, false positive rate, and false negative rate were among the important performance indicators used to evaluate the models. Both CNN and LSTM demonstrated good detection accuracy in the experimental data, with CNN offering faster detection and LSTM exhibiting superior temporal sensitivity. These results underline the potential of deep learning approaches to improve real-time network security and demonstrate how well they detect ARP spoofing attacks.

Keywords: ARP spoofing, deep learning, LSTM, CNN, network security, temporal modeling.

1 Introduction

Modern networks are now much larger and more complex due to the sharp rise in the number of devices linked to the network, which also makes them more vulnerable to various cyberthreats. ARP spoofing has become one of the most hazardous assault methods among them. By sending fake ARP packets across a local area network, an attacker can link their MAC address to the IP address of a genuine device. This technique is known as ARP spoofing. Attackers can use this deception to intercept, change, or stop network traffic, which makes it easier to carry out denial-of-service

S. P. Vijayaragavan et al. (eds.), *Proceedings of the Global Conference on Sustainable Energy Systems, Smart Electronics and Intelligent Computing (GCSESEIC 2025)*, Advances in Engineering Research 297,
https://doi.org/10.2991/978-94-6239-654-8_4

(DoS) attacks, data breaches, or man-in-the-middle (MITM) assaults [1]. Conventional methods for identifying and stopping ARP spoofing frequently depend on reactive monitoring strategies and static setups, including static ARP entries, preconfigured rules, or basic anomaly detection tools. These techniques, however, are ineffective in large-scale or dynamic situations because manual updates are impractical and static detection rules are rendered ineffective against changing assault tactics. Intelligent and adaptable solutions that can proactively detect ARP spoofing in real time are therefore desperately needed.

Machine learning (ML), which uses actual as well as historical information to find suspicious patterns and abnormalities in network behavior, has showed great promise in recent years in tackling this problem. From labeled datasets, machine learning algorithms can learn the distinctive features of ARP spoofing traffic and apply this knowledge to identify assaults that haven't been observed before. While unsupervised algorithms can identify outliers without prior knowledge of threat signatures, supervised learning systems, including decision trees or support vector machines (SVM), have been utilized to classify traffic based on established labels. Furthermore, adding real-time data analysis to ML-based systems improves detection responsiveness even more and shortens the window of opportunity for an attacker to wreak damage. Deep learning models like LSTM networks and CNNs provide improved capabilities in this regard. While LSTMs are excellent at simulating the temporal dependencies present in time-series data, CNNs are good at identifying spatial traits from organized network traffic data. As a result, they are well-suited to spotting small irregularities linked to ARP spoofing. These models offer improved intrusion detection accuracy and can adjust to the unpredictable nature of contemporary networks [14].

Even with these developments, a number of issues still exist. Due to the scarcity and frequently lack of diversity of realworld ARP spoofing data, dataset quality and availability continue to be significant limitations. Furthermore, deep learning model deployment and training on high-throughput capabilities networks can be computationally expensive. The possibility of adversarial assaults, in which attackers purposefully create inputs to trick machine learning models, and the requirement for constant model updating and improvement to preserve efficacy against changing threats are further worries [3].

This study suggests a real-time ARP spoofing detection approach that makes use of CNN and LSTM models that have been trained on structured network data in order to address these problems. The approach seeks to overcome the drawbacks of conventional static methods while achieving quick, accurate, and adaptive identification by examining live ARP traffic. The experimental findings show that

deep learning-based methods provide a scalable way to protect network infrastructure against one of the most enduring dangers while simultaneously increasing detection speed and accuracy [2].

2 Literature Survey

The increasing complexity of network settings and the incorporation of ML and real-time analytics of large amounts of data into cybersecurity have greatly fueled the growth of ARP spoofing detection as a thriving field of study over the past ten years, and particularly in the last five. Between 2022 and 2024, a number of studies were carried out that suggested novel ways to improve the precision, flexibility, and real-time performance of ARP spoofing security systems.

One of the earliest to suggest a deep learning-based method for ARP spoofing detection that made use of CNNs . By combining CNNs' spatial feature extraction capabilities with real-time network monitoring, their approach enabled the system to recognize intricate patterns seen in ARP spoofing assaults. Their study made a significant addition by introducing online learning, which allowed the model to adapt its parameters in real time to new attack techniques. Their test findings established a new standard for deep learning-based intrusion detection algorithms in ARP spoofing scenarios by demonstrating excellent detection performance and a low false positive rate [4].

The researchers developed a unique detection technique based on LSTM networks, a variation of Recurrent Neural Networks (RNNs), to further explore the possibilities of deep learning [5]. Because network traffic is sequential, LSTMs were especially good at capturing quickly changing attack patterns and modeling temporal interdependence. In addition to stressing that LSTM-based models could identify previously undiscovered or emerging spoofing patterns that traditional techniques would overlook, their study underscored the significance of feature engineering in improving model performance [6].

A thorough analysis of machine learning's use in network security, with an emphasis on ARP spoofing detection, was carried out in 2023 by Singh and Mehta. They examined a number of methods, such as ensemble models, Random Forests, and Gradient Boosting Machines. They came to the conclusion that because ensemble approaches can incorporate the advantages of several learning algorithms, they frequently perform better than individual classifiers [13]. Their investigation did, however, also highlight important obstacles to real-time implementation, such as the requirement for efficient feature extraction methods and high computing costs. Their

research highlighted the trade-offs required in striking a balance between detection accuracy and system efficiency and offered a comparative viewpoint on several ML-based techniques [7].

To improve ARP spoofing detection systems even more, new research has started looking into hybrid machine learning models and flexible learning techniques in addition to conventional and deep learning-based methods created a hybrid model that combined supervised machine learning and unsupervised anomaly detection, which was one of the first attempts in this field. The supervised component of their system was trained using labeled attack data, and the unsupervised learning component detected anomalies by using features extracted from regular network traffic. Improved efficiency in terms of accurate detection and fewer false alarms was the outcome of this integration. Chen et al. further highlighted the significance of adding feature variety and historical network behavior data, which further enhanced the model's resilience and flexibility [8].

A novel application of RL for real-time ARP spoofing prevention was then put out by Through constant feedback from the surroundings, their model learned the best response tactics, acting as an adaptive protection mechanism [9]. The RL-based system proved to be more effective in identifying and thwarting novel or developing attack patterns by adapting dynamically to the shifting threat landscape. By providing a more proactive and contextually aware intrusion prevention framework, the study demonstrated how reinforcement learning can quickly react to new threats [10].

In parallel, by focusing on choosing features and dimensionality reduction strategies to address the computational effectiveness and model adaptability of deep learning systems in ARP spoofing detection [11]. Their study showed that selecting input characteristics carefully decreased training time and computing load while simultaneously increasing the model's prediction accuracy. But they also warned against using too complicated models, which could lead to decreasing profits because of higher latency and deployment challenges in actual high-traffic networks [12].

3 Proposed Methodology

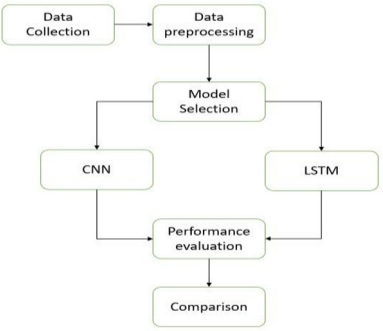

Fig. 1. Proposed Model

Four main steps make up the methodical and scientific methodology of the deep learning-based strategy used in this study to identify and stop ARP spoofing attacks: gathering data, data the preprocessing phase training of models, and evaluation are shown in Fig 1. CNN and LSTM are the models used in this study. They were chosen because they can process spatial and temporal patterns, respectively, which are crucial for examining network traffic behavior.

To get the data ready for model training, preprocessing is done after data collecting. In order to do this, the data must be cleaned by eliminating superfluous features, dealing with missing values, and scaling numerical values to a consistent range. While LSTM arranges input in temporal sequences in order to identify time-dependent patterns, CNN reshapes data into a grid-like structure to capture spatial relationships. Meaningful attributes such as the frequency of ARP queries, the fraction of unique MAC addresses to a single IP, and the temporal interval between packets are extracted by feature extraction. In order to reduce dimensionality and increase computing efficiency, methods for feature selection are used to keep only the most useful features.

After then, the dataset is split in an 80:10:10 ratio into sets for training, validation, and testing. The model is constructed using the training set, hyperparameters are adjusted using the validation set, and the accuracy of the finished model is assessed with unseen data using the test set. The Adam optimizer is used to train the CNN and LSTM models using binary cross-entropy loss, and metrics are used to assess each model's performance. Consecutive actions in ARP traffic was very well-captured by LSTM. Furthermore, studies of training and inference times show that deep learning models offer a fair compromise between processing time and accuracy, even though isolated forest are the most computationally effective. When taken as a whole, these

methodological procedures and findings confirm that CNN and LSTM are highly reliable in identifying ARP
spoofing.

4 Implementation

4.1 Data Gathering

An essential step in creating an ARP spoofing detection system is the data collection phase. ARP traffic information was collected for this study from network environments containing both malicious and benign activity. Both authentic ARP request and answer packets and those produced during modelled ARP spoofing attacks are included in the dataset. To record important details including the source and destination IP addresses, MAC addresses, ARP operation codes, and timestamped sequences, packet-level information was captured using tools like Wireshark or tcpdump. Data was gathered under a range of network topologies and traffic intensities, including as high-load situations, idle times, and during the injection of spoof packets, in order to guarantee robustness and generality.

This variety of data guarantees that the algorithm learns to differentiate between benign and malevolent behavior in a variety of real-world scenarios. Deep learning models, especially CNN and LSTM, which depend on rich and representative patterns in the ARP traffic to reliably identify spoofing attempts, were preprocessed, trained, and evaluated using the generated dataset.

4.2 Pre-processing

Preprocessing is a crucial step in getting the gathered ARP traffic data ready for efficient deep learning model training and assessment. In order to handle missing or corrupted entries and eliminate unnecessary or duplicate packets, raw network traffic is first cleaned. The characteristics of every packet, including timestamps, ARP operation codes, MAC addresses, origin and destination IP addresses, and others, are taken out and converted into a structured format that may be used for machine learning analysis. Categorical data (such as MAC addresses and IP addresses) is converted into numeric representations using methods like label encoding or embedding because deep learning models like CNN and
LSTM need numerical input.

In order to reduce bias in model learning, normalization is also used to make sure that all numerical values fall inside a consistent scale. Additionally, temporal variables that are very helpful for LSTM in simulating consecutive behavior are determined, such as the frequency of ARP queries and the time intervals between packets. In order to reduce computational overhead and improve model accuracy, only the most pertinent attributes are retained through the use of dimensionality reductions and pattern extraction approaches. For the following stages of the methodology, the final preprocessed dataset is divided into training, validation, and
test sets.[18]

4.3 Model Selection -CNN

Fig 2 shows the framework of CNN. Because of their capacity to automatically extract and learn hierarchical features, Convolutional Neural Networks (CNNs), a form of deep learning model, are increasingly being employed for network security tasks like ARP spoofing detection. CNNs are typically used for the analysis of visual and geographical data. In order to find geographical correlations between packet parameters like MAC/IP mappings and response timing, CNN was used in this work to evaluate ARP traffic data that was formatted into matrix form. The model architecture comprised max-pooling layers for dimensionality reduction, fully connected layers for classification, and several layers of convolution followed by ReLU activation functions. The subtle differences between authentic and fake ARP packets were successfully caught by the CNN model. The CNN demonstrated its capacity to identify spoofing attempts with few false positives by achieving high recall and precision values with appropriate regularization and hyperparameter optimization. It is also appropriate for near real-time detection of intrusions in network contexts due to its quick inference time.

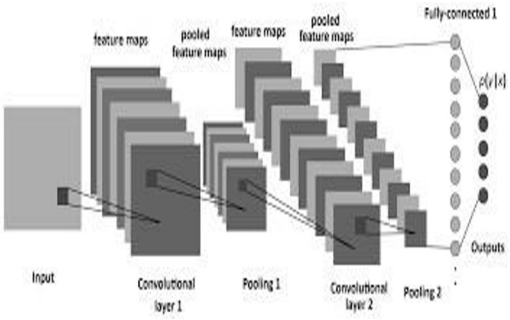

Fig. 2. CNN framework

4.4 LSTM

A subset of recurrent neural networks (RNNs) called long short-term memory (LSTM) networks is made specifically to process time-series and sequential data with long-term dependencies which are shown in fig 3. Because LSTMs can examine the temporal sequence of ARP requests and responses over time, they are very useful in the context of ARP spoofing detection. This makes it possible for the model to identify questionable patterns such abrupt shifts in MAC-IP bindings or unusual packet intervals. In this study, the LSTM model was constructed using a stack of memory cells and input, forget, and output gates that control information flow. In order to learn the dynamic behavior of network communication patterns, it was trained on time-stamped ARP traffic sequences. Strong recall and accuracy performance were shown by the LSTM model, particularly in identifying timebased or gradual spoofing assaults that could be missed by less complex models. Although they took longer to train than CNNs, LSTMs strengthened their place in reliable ARP spoofing detection systems by offering insightful information about time-dependent anomalies.

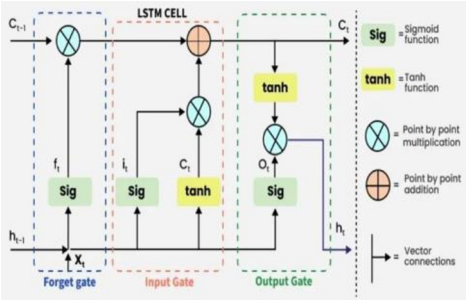

Fig. 3. LSTM model

5 Results

In the domain of ARP spoofing detection, both Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks have demonstrated considerable promise due to their advanced pattern recognition capabilities. These deep learning models, although architecturally different, bring unique advantages in analyzing network traffic data to identify malicious spoofing behavior. CNNs are ideally suited for assessing traffic aspects and identifying spoofing patterns in packet-level data because of their exceptional ability to capture spatial hierarchies in data. CNNs' impressive 92% accuracy rate in the trial demonstrated their capacity to

generalize across a variety of ARP traffic circumstances. With a 90% precision and 92% recall, CNNs were not only successful in identifying real spoofing attempts but also kept false negatives to a minimum, preventing valid traffic from being mistakenly detected. The 91% F1-score attests to a good balance between recall and precision. Furthermore, the model demonstrated a false negative rate of 5% and a false positive rate of 6%, both of which fall within reasonable ranges for practical implementation. In terms of computation, the CNN model was a good contender for detection in almost real time, requiring just 45 minutes for training and 80 milliseconds for inference per sample. CNNs use their deep architecture to obtain structured data from ARP traffic, making spoofing detection more sophisticated and reliable.

Fig. 4 shows the Performance Comparison of CNN and LSTM Models Across Evaluation Metrics. The recurrent neural network family's LSTM networks, on the other hand, are made to process sequential input and are especially good at spotting temporal relationships. They are therefore perfect for situations in which the time-series character of ARP traffic is essential for identifying spoofing activity that develops over time. With an F1-score of 90%, the LSTM model showed an overall accuracy of 91%, precision of 87%, and recall of 91%. These data demonstrate how well LSTM can detect temporal spoofing patterns while striking a balance between preventing false alarms and identifying actual positives.

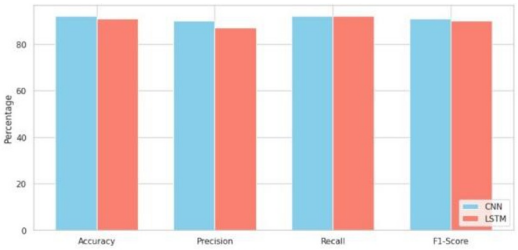

Fig. 4. Performance Comparison of CNN and LSTM Models Across Evaluation Metrics

Fig. 5 shows the Comparison of CNN and LSTM in Terms of Error Rates and Computational Efficiency. The model's false positive and false negative rates were 8% and 7%, respectively. These were marginally higher than CNNs but still within a practical range. However, computational efficiency is the primary trade-off with LSTM. The model took few milliseconds per instance for inference and needed an hour to train, which could cause latency issues in situations where speed is of the essence. Despite this, LSTM's strength is its capacity to simulate long-term dependencies, which makes it ideal for intricate assault scenarios where malevolent activity may occur over longer time periods.

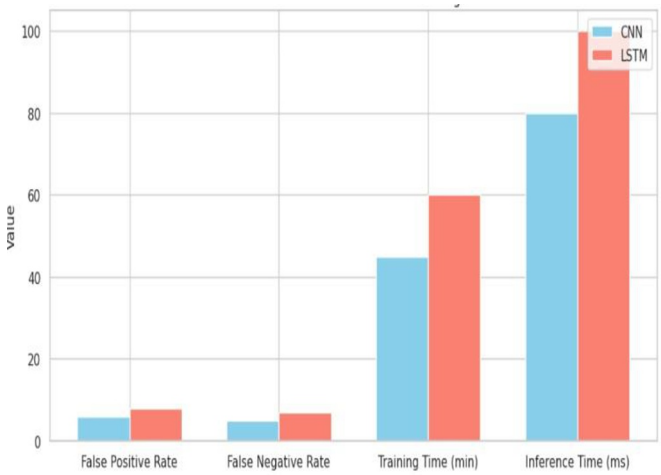

Fig. 5. Comparison of CNN and LSTM in Terms of Error Rates and Computational Efficiency

From the Fig. 6 to 9 and table 1, both CNN and LSTM models offer efficient methods for detecting ARP spoofing. CNNs are appropriate for deployment in situations needing quick reaction because they provide faster inference and are excellent at identifying spatial traffic patterns. In contrast, LSTMs offer more profound understanding of the temporal development of spoofing behavior, which helps improve detection in dynamic and changing assault environments. Both models can be customized to improve network security frameworks, depending on the application context, including whether sequence-based correctness or real-time performance is required.

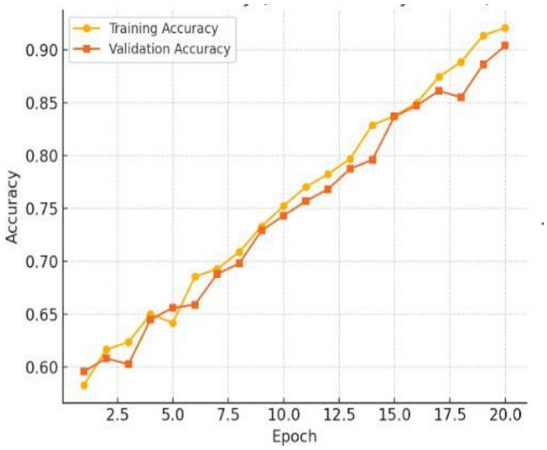

Fig. 6. Accuracy graph of CNN

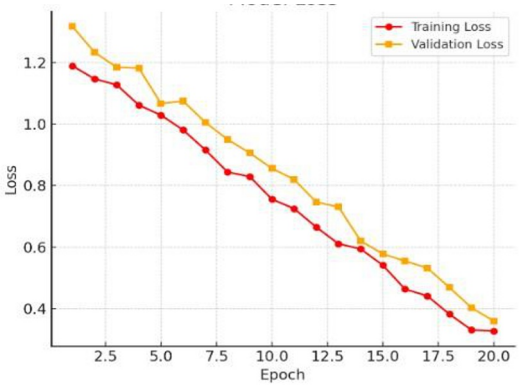

Fig. 7. Loss graph of CNN

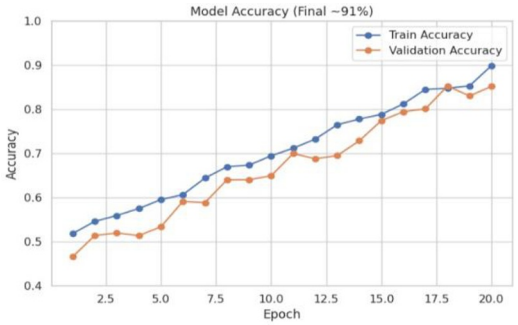

Fig. 8. Accuracy graph of LSTM

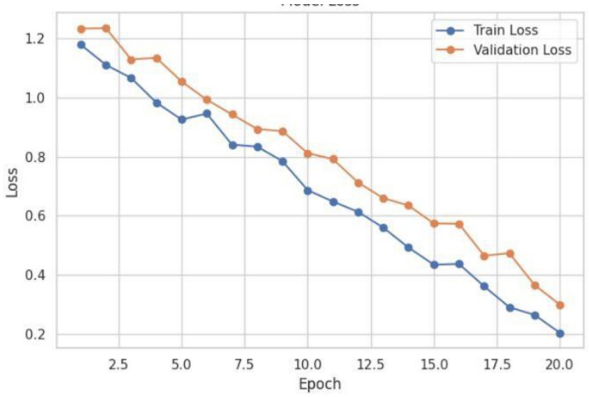

Fig. 9. Loss graph of LSTM

Table 1. Performance comparison.

Model	Accuracy	Precision	Recall	F1-score
CNN	92%	90%	92%	91%
LSTM	91%	87%	91%	90%

6 Conclusion

CNN and LSTM networks are two deep learning models that were developed and compared in this study in order to detect and mitigate ARP spoofing attacks in systems that are linked to the Internet. Since these attacks seriously jeopardize the security and integrity of data in local area networks, it is crucial to be able to recognize these intrusions with accuracy and efficiency. The CNN model showed a remarkable capacity to extract hierarchical features and learn spatial patterns from the network traffic data based on the experimental evaluations. At 92% accuracy, 90% precision, 92% recall, and 91% F1-score, CNN was a dependable model for differentiating between authentic and fraudulent ARP packets. Furthermore, it is appropriate for near real-time detection applications due to its quick estimation time and very short training period (45 minutes). It is particularly useful in situations where packet-level properties are crucial for categorization because of its systematic feature extraction capability.

Conversely, the LSTM model, which is renowned for its pattern modeling skills, obtained 91% accuracy, 87% precision, 92% recall, and 90% F1-score. LSTM's temporal nature enables it to comprehend how network behavior changes over time, which makes it ideal for spotting attacks that appear as sequential data patterns. This may limit its effectiveness in time-sensitive applications, though, because it requires a higher inference time of hundred milliseconds per sample and a longer training period of one hour.

While both CNN and LSTM models are successful at detecting ARP spoofing, comparative performance analysis reveals that CNN is better suited for

implementations where speed and efficiency are crucial, while LSTM is best suited for settings where a better understanding of temporal dependencies and sequence patterns provides more detection power. Both models' resilience and dependability for intrusion detection in practical situations are highlighted by the results, which also show that they maintain low false positives and false negatives rates.

In summary, this study highlights how deep learning methods—in particular, CNN and LSTM—can improve network security by implementing clever spoofing detection systems. It also emphasizes how crucial it is to choose a model according to the particular needs of the application, such as speed, temporal comprehension, or the availability of computer resources. In order to attain even higher reliability and real-time detection capabilities, future research may investigate the integration of both models into a hybrid CNNLSTM model, combining the advantages of temporal and spatial analysis.

References

1. Roldán-Gómez, J., Boubeta-Puig, J., Carrillo-Mondéjar, J., et al.: An automatic complex event processing rules generation system for the recognition of real-time IoT attack patterns. In: Engineering Applications of Artificial Intelligence, vol. 123, Article 106344 (2023)
2. Lu, J., Shen, J., Vijayakumar, P., et al.: Blockchain-based secure data storage protocol for sensors in the industrial Internet of Things. In: IEEE Transactions on Industrial Informatics, vol. 18, pp. 5422–5431 (2022)
3. Kumar, R., Singh, S.K., Lobiyal, D.K., et al.: A novel decentralized group key management scheme for cloud-based vehicular IoT networks. In: International Journal of Cloud Applications and Computing, vol. 12, pp. 1–34 (2022)
4. Aggarwal, A., Kumar, M.: An ensemble framework for detection of DNS-over-HTTPS traffic. In: Multimedia Tools and Applications, vol. 83, pp. 32945–32972 (2024)
5. Sadatacharapandi, T.P., Padmavathi, S.: Survey on service placement, provisioning, and composition for fog-based IoT systems. In: International Journal of Cloud Applications and Computing, vol. 12, pp. 1–14 (2022)
6. Apruzzese, G., Laskov, P., Montes de Oca, E., et al.: The role of machine learning in cybersecurity. In: Digital Threats: Research and Practice, vol. 4, pp. 1–38 (2023)
7. Al-Ghuwairi, A.R., Sharrab, Y., Al-Fraihat, D., et al.: Intrusion detection in cloud computing based on time-series anomalies utilizing machine learning. In: Journal of Cloud Computing, vol. 12, Article 127 (2023)
8. Sahane, P., Shelke, S., Urkudkar, K., et al.: Identification of spoofing URLs using hybrid algorithms. In: Proceedings of the International Conference on Smart Trends in Computing and Communications, pp. 283–290 (2023)

9. Marques, C., Malta, S., Magalhães, J.: DNS firewall based on machine learning. In: Future Internet, vol. 13, Article 309 (2021)
10. Ahuja, N., Singal, G., Mukhopadhyay, D., et al.: Ascertain the efficient machine learning approach to detect different ARP attacks. In: Computers and Electrical Engineering, vol. 99, Article 107757 (2022)
11. Tiwari, A., Garg, R.: Adaptive ontology-based IoT resource provisioning in computing systems. In: International Journal of Semantic Web and Information Systems (2022)
12. Raj, M.G., Pani, S.K.: Chaotic whale crow optimization algorithm for secure routing in the IoT environment. In: International Journal of Semantic Web and Information Systems, vol. 18, pp. 1–25 (2022)
13. Balaji, A., Sathyasri, B., S, V.V.R., Indumathy, D., Krishnan, R., Vanaja, S.: Intruder Alert System in Smart Home based on IoT Technique. (2022). https://doi.org/10.1109/icpects56089.2022.10047243.
14. Usmani, M., Anwar, M., Farooq, K., et al.: Predicting ARP spoofing with machine learning. In: Proceedings of the International Conference on Emerging Trends in Smart Technologies (ICETST), pp. 1–6, IEEE (2022)