

Enhancing network security using unsupervised learning approach to combat zero-day attack

Rajakumar Perumal¹, Tamilarasi Karuppiyah², Uppiliraja Panneerselvam¹, Venkatesan Annamalai³, Prabu Kaliyaperumal¹

¹School of Computer Science and Engineering, Galgotias University, Greater Noida, India

²Department of Information Technology, Panimalar Engineering College, Varadarajapuram, India

³Department of Electrical, Electronics and Communication Engineering, Galgotias University, Greater Noida, India

Article Info

Article history:

Received Apr 11, 2024

Revised Aug 6, 2024

Accepted Aug 15, 2024

Keywords:

Autoencoder

Cloud security

Deep learning

Intrusion detection system

Support vector machine

ABSTRACT

Machine learning (ML) and advanced neural network methodologies like deep learning (DL) techniques have been increasingly utilized in developing intrusion detection systems (IDS). However, the growing quantity and diversity of cyber-attacks pose a significant challenge for IDS solutions reliant on historical attack signatures. This highlights the industry's need for resilient IDSs that can identify zero-day attacks. Current studies focusing on outlier-based zero-day detection are hindered by elevated false-negative rates, thereby constraining their practical efficacy. This paper suggests utilizing an autoencoder (AE) approach for zero-day attack detection, aiming to achieve high recall while minimizing false negatives. Evaluation is conducted using well-established IDS datasets, CICIDS2017 and CSECICIDS2018. The model's efficacy is demonstrated by contrasting its performance with that of a one-class support vector machine (OCSVM). The research underscores the OCSVM's capability in distinguishing zero-day attacks from normal behavior. Leveraging the encoding-decoding capabilities of AEs, the proposed model exhibits promising results in detecting complex zero-day attacks, achieving accuracies ranging from 93% to 99% across datasets. Finally, the paper discusses the balance between recall and fallout, offering valuable insights into model performance.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Rajakumar Perumal

School of Computer Science and Engineering, Galgotias University

Greater Noida, Uttar Pradesh, 201310, India

Email: rajkumar.jcet@gmail.com

1. INTRODUCTION

Critical to addressing the exponential increase in cyber-attacks [1], [2] is the deployment of intrusion detection systems (IDS) capable of identifying zero-day cyber-attacks. Machine learning (ML) methodologies have been widely employed in developing resilient IDS [3], [4] solutions. As depicted in Figure 1, IDS stands as a vital component of modern cybersecurity [5], fortifying networks against escalating digital risks. However, contemporary IDS systems excel in detecting known attacks but struggle with identifying novel zero-day attacks [6] due to reliance on predetermined patterns and signatures. Additionally, elevated false-positive rates limit their effectiveness and practicality in real-world deployments. Consequently, a notable portion of zero-day attacks evade detection, exacerbating their repercussions such as denial of service and theft of customer information. According to Ali *et al.* [7], a zero-day attack is a traffic pattern that is of interest and often lacks matching patterns in malware or network attack detection elements. Sakthimurugan *et al.* [8] delve into the real-world ramifications of zero-day attacks. Their research is

centered on examining the impact and prevalence of these attacks. The authors emphasize that zero-day attacks are considerably more widespread than anticipated. Their analysis of 18 attacks reveals that 11 were previously unidentified. Their discoveries indicated that zero-day attacks may endure over a substantial period, with an average of 10 months, before detection. This prolonged presence allows them to compromise systems over the said period. Moreover, Zoppi *et al.* [9] cite a statistical investigation revealing that 62% of attacks are detected only after systems have been successfully compromised. Furthermore, the count of zero-day attacks in 2023 surpasses that of the preceding three years [10].

One primary research avenue for detecting zero-day attacks revolves around identifying outliers, which are instances that deviate from normal network traffic. However, the main limitation of current detection techniques based on outliers lies in their relatively low accuracy rates, stemming from both high false positive rate (FPR) and false negative outcomes. As mentioned earlier, the elevated false-negative rates (FNR) make the setup vulnerable to attacks [11], whereas the elevated FPR unnecessarily burden cybersecurity operation centers, with only 32% of investigated intrusions proving to be genuine. Essa and Bhaya [12] underscore the limitations posed by false negatives in IDS development, notably their adverse impact on IDS effectiveness.



Figure 1. Components of intrusion detection system

Hairab *et al.* [13] introduced a system tailored to identify previously unseen attacks within interconnected device networks. Their approach relies on a distributed detection mechanism. They utilized artificial neural networks for supervised anomaly-based ML, attaining a detection rate of 94% on the NSL-KDD dataset. However, the study's limitation is the relatively small dataset used for supervised learning. Sun *et al.* [14] introduced a Bayesian probabilistic approach with the objective of detecting zero-day attack pathways. They represented attacks within a structure resembling a graph and presented a prototype aimed at identifying these attacks. Duong and Hai [15] proposed a semi-supervised ML approach for zero-day attack detection utilizing the NSL-KDD dataset. They utilized a K-means clustering model to remove outliers during training, aiming for a training dataset with only benign network traffic. Despite feature discrepancy, their method achieved a 91% detection accuracy.

This paper advocates leveraging unsupervised learning capabilities for outlier detection to effectively identify previously unknown attacks with a high rate of accuracy. The goal entails crafting an intrusion detection model that is efficient at identifying novel intrusions and zero-day attacks with a high accuracy, thereby ensuring a low fallout rate. Thus, possessing a strong ability to detect zero-day attacks can alleviate the challenges and concerns typically linked with novel attacks.

This work offers three significant contributions:

- a) Introducing and deploying an innovative and efficient autoencoder (AE) framework for zero-day attack detection in intrusion detection system.
- b) Creating an anomaly detection model utilizing one-class support vector machine (OCSVM).
- c) Assessing the performance of the OCSVM model, utilized as a foundational outlier-based detector, compared with the suggested AE framework, while also comparing the proposed model with benchmarked algorithms.

2. THE COMPREHENSIVE THEORETICAL BASIS

2.1. Autoencoder

The model presented in this manuscript primarily leverages the features of AE. The objective is for the AE to operate as a lightweight anomaly detector [16], thus facilitating zero-day attack detection. AEs were initially introduced by Rumelhart *et al.* [17] to address backpropagation challenges in an unsupervised context by employing the input as the target. AEs are classified as self-supervised models because their input and output are essentially the same. Brunner *et al.* [18] explained that an AE is a neural network variant that, through training, learns to produce output resembling its input. Figure 2 depicts the basic structure of an AE.

The setup of an AE and its latent layer count vary depending on the domain and usage context. The re-construction error is computed using a function that measures the disparity between the input and the

output, where the output is the reconstructed input [19]. Mean square error (MSE) and mean absolute error (MAE) are typical functions employed to compute the reconstruction error [20]. By training the AE to minimize reconstruction error, it enables precise differentiation between normal and malicious traffic. The architecture depicted in Figure 2 showcases the effectiveness of the AE model in extracting crucial information, leading to robust classification results [21].

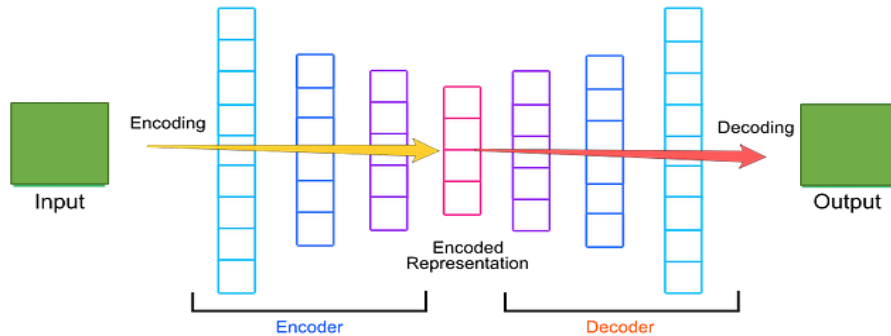


Figure 2. Architecture of an AE

2.2. One-class support vector machine

Support vector machine (SVM) is renowned as a deeply rooted and robust supervised ML method [22]. With the provided training data, an SVM is trained to define a hyperplane in a multi-dimensional space, aiming to effectively differentiate between the classes [23]. In the case of two dimensions, this hyperplane is represented as a line, while in three or higher dimensions, it manifests as a plane [24]. When data exhibits non-linear separability, a kernel is utilized to transform the input features into a higher-dimensional space, enabling better separation [24]. In this transformed space, a hyperplane can accurately distinguish between the classes [23].

The OCSVM differs from its supervised counterpart by functioning as an unsupervised variation. It is characterized as a system adept at identifying "novelty" [22]. The goal of the OCSVM is to define a hyperplane [25], illustrated in Figure 3, that acts as a boundary. This boundary effectively encompasses all training data points while excluding any others. The outcome of training an OCSVM is observed as a boundary with a spherical shape [22]. Given its reputation as a highly established outlier-based ML techniques, the OCSVM serves as an optimal benchmark for assessing the effectiveness of a deep neural network (DNN)-based AE.

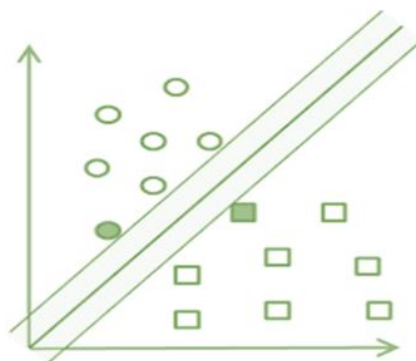


Figure 3. Visualization of the OCSVM hyperplane

3. RESEARCH METHOD

The proposed detection model encompasses several stages. It begins with preprocessing traffic data, followed by conducting dimensionality reduction, training the detection model, identifying anomalies, and evaluating its performance. The architecture illustrated in Figure 4 elucidates the design of the attack

detection method. Initially, traffic data undergo normalization and standardization to prepare normal network traffic datasets. Subsequently, implementing unsupervised pre-clustering with AE produces distinct subsets within the initial latent layer, highlighting unique patterns while reducing dimensions. During the training phase, these subsets are individually processed through the same AE in latent spaces 2 to 4, and OCSVM. The "encoding-decoding" process in AE evaluates the average reconstruction error, determining the detection threshold. In the anomaly detection phase, the trained model utilizes test data to generate reconstructed outputs, with the reconstruction error calculated using the mean squared error. If the calculated error surpasses the predefined threshold, the traffic is identified as zero-day attack; otherwise, it is classified as normal [18]. OCSVM operates within a high-dimensional kernel space by transforming samples from reduced dimensions using a mapping function.

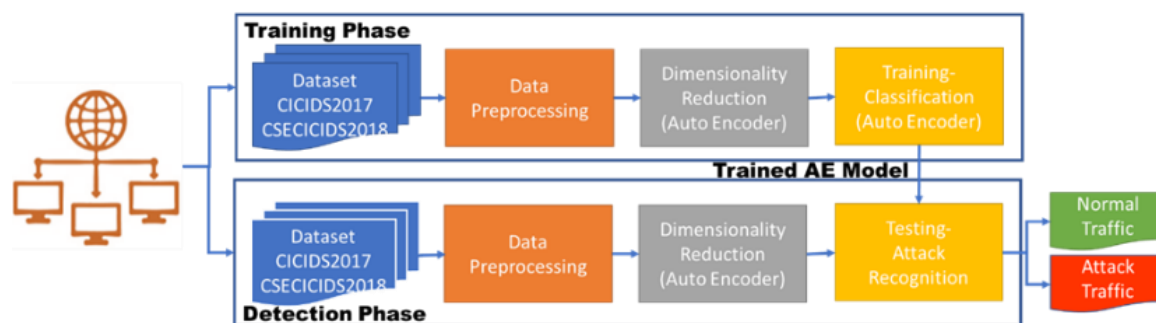


Figure 4. Architecture of proposed autoencoder based IDS

3.1. Preprocessing data and reducing dimensionality

This study makes use of benchmark datasets for intrusion detection, such as CICIDS2017 and CSECICIDS2018. Standard scaling techniques are employed to standardize numerical data, linearly transforming it. The normalized data is subsequently transformed to a range with a mean (μ) of 0 and a standard deviation (σ) of 1 through Z-score normalization.

The AE, a neural network designed for unsupervised learning and dimensionality reduction, handles input normal traffic data via its encoder network. Every layer employ weights and biases to modify the data into a representation of lower dimensionality, ultimately resulting in the latent layer generating an encoded representation. The complete network, including both the encoder and decoder, experiences end-to-end training through the utilization of backpropagation and gradient descent algorithms. Through this unsupervised learning process, a compressed representation of the input data is obtained, effectively capturing its fundamental features. Having been trained on unlabeled data, the AE's encoded representation acts as a lower-dimensional and pre-clustered variant of the original normal data, thereby enabling efficient analysis and anomaly detection.

3.2. Detecting zero-day attacks with AE

The AE, a neural network trained without supervision, extracts hierarchical features to enhance classification. By leveraging the latent space 1 within the AE for dimensionality reduction, this study constructs an AE model to effectively capture data features across clustering subsets. Training the AE reduces the reconstruction error, enabling precise discrimination between normal and malicious traffic. The architecture depicted in Figure 4 showcases the effectiveness of the AE model in extracting vital information to achieve robust classification results.

During the training phase, the AE model processes batches of normal traffic data with the objective of minimizing the reconstruction error. The detection threshold dynamically adjusts based on the cumulative loss incurred during the model's training process. The process iterates for a designated number of epochs, yielding the trained AE model and detection threshold as outputs. During the detection phase, the trained AE model is assessed using the test dataset to detect anomalies. The reconstruction error for each data point is determined by comparing the original and reconstructed data. Data points exceeding the detection threshold in reconstruction error are classified as zero-day attack; otherwise, they are labeled normal. The output obtained indicates the classification of each data point in the test dataset as either normal or zero-day attack. This testing process is crucial for evaluating the effectiveness of the trained detection model on new data, assisting in the identification of potential anomalies.

3.3. Detecting zero-day attacks with OCSVM

OCSVM stands out as a unique approach in one-class classification, focusing solely on a singular data class [22], unlike binary or multi-class methods. After training, the model determines whether a new sample belongs to the designated target class. OCSVM operates in a high-dimensional kernel space by transforming samples using a mapping function. It formulates a hyperplane to separate data from the origin, addressing a quadratic problem with a weight vector and margin. The hyperparameter ' ν ' controls the trade-off, limiting the proportion of anomalies. The inclusion of non-negative slack variables allows for soft margins, adjusting sensitivity to outliers. The Gaussian kernel function effectively represents data, with the hyperparameter ' γ ' shaping the decision boundary. Zero-day attacks are identified when new samples fall on the incorrect side of the hyperplane, resulting in negative values in the decision function.

4. RESULTS AND DISCUSSION

The suggested methods are evaluated using benchmark intrusion detection datasets, with a specific focus on the top four attacks as zero-day occurrences within both the CICIDS2017 and CSECICIDS2018 datasets. The evaluation involves comparing the proposed model against standard anomaly detection algorithms OCSVM on un-clustered data, followed by an assessment of their performance. The Table 1 presents the dataset utilized and the quantity of selected samples for this research.

Table 1. Experimental samples

	CICIDS2017	CSECICIDS2018
Training class	Benign	Benign
Testing class	DoS-DDoS-Brute Force	Botnet-Benign
Features	83	80
Benign	2,359,087	2,374,871
Attack	224,893	239,842

4.1. Performance metrics

Evaluating the efficiency of an intrusion detection model using a confusion matrix [26] involves assessing accuracy, recall, and specificity. Recall, as depicted in (1), represents the accurate identification of positive instances. Specificity, outlined in (2), gauges the system's capability to correctly recognize true negative instances, serving as a comprehensive evaluation metric. Accuracy, as indicated in (3), measures the precision of classifications.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (1)$$

$$Specificity = \frac{True\ Negative}{True\ Negative + False\ Positive} \quad (2)$$

$$Accuracy = \frac{True\ Positive + True\ Negative}{Predicted\ Positive + Predicted\ Negative} \quad (3)$$

The successful performance of the model hinges on critical factors, including the suitable architecture and hyperparameter configurations for the AE detection model. Extensive experimentation was undertaken to identify the most effective hyperparameter configurations. The AE proposed in the method comprises an input layer, five latent space, and an output layer. Additionally, the AE integrates three hidden layers, with unit sizes customized according to the loss function. This approach guarantees an optimal adaptation to the diverse feature dimensions within the training data.

4.2. Training and testing the detection model

This approach entails dividing the dataset into training (70%) and testing (30%) subsets comprising benign data. All attack data is utilized for evaluation purposes. The model undergoes training with early stopping criteria to ascertain the ideal number of epochs. Rectified linear unit (ReLU) is employed as the activation function, while Adam optimization with a learning rate set at 0.0001 is utilized. Detection accuracy is evaluated by employing MSE as the loss function. Figure 5 illustrates the testing accuracy of the proposed approach across both datasets, emphasizing the efficacy of the model. Detection models constructed using both Normal and Attack data from the CICIDS2017 and CSECICIDS2018 datasets demonstrate significant performance across all datasets. The experimental results of the AE model, averaged across ten experiments, demonstrate significant performance across both datasets.

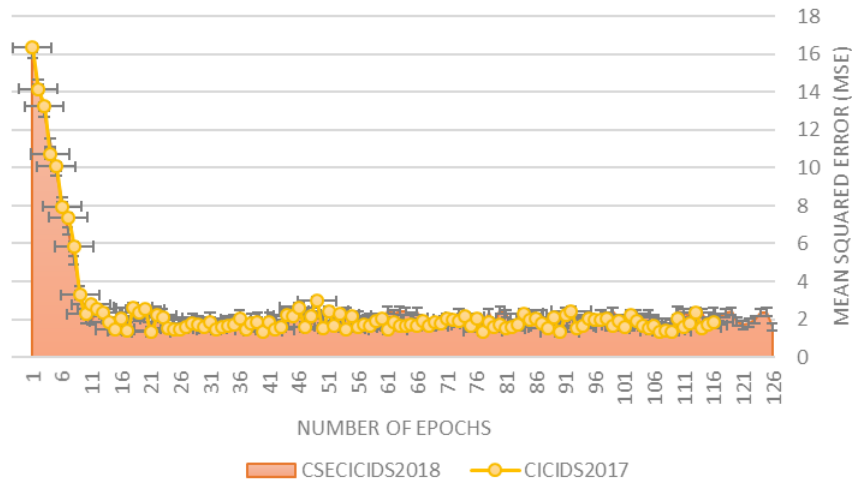


Figure 5. Proposed AE loss function

4.3. Assessing the AE on the CICIDS2017 and CSECICIDS2018 datasets

As stated earlier, the AE is trained using 70% of the benign instances. The optimized architecture of the AE for the CICIDS2017 and CSECICIDS2018 datasets consists of a neural network comprising 19 neurons in latent space 1 and 5, along with three hidden layers consisting of 14, 10, and 14 neurons. The ideal batch size for optimal performance is 1,024. Along with other optimized parameters such as a mean square error loss (MSE) function, L2 regularization set at 0.0001 and training durations of 114 epochs for CICIDS2017 and 126 epochs for CSECICIDS2018 datasets, respectively.

Table 2 provides a summary of the AE accuracy for all classes in the CICIDS2017 and CSECICIDS2018 datasets. Accuracy differs for benign and attack classes. For benign, it reflects specificity, indicating instances correctly classified as non-zero-day. For attacks, it represents recall, capturing correctly identified instances.

Table 2. AE accuracy for all classes in the CICIDS2017 and CSECICIDS2018 datasets with different threshold values

Threshold/attack	CICIDS2017			CSECICIDS2018		
	0.15	0.1	0.05	0.15	0.1	0.05
Benign	0.9659	0.9529	0.9147	0.9889	0.9598	0.9248
DoS	0.9842	0.9858	0.9887	0.9877	0.9878	0.9898
DDoS	0.9823	0.9898	0.9899	0.9882	0.9881	0.9891
Brute Force	0.9301	0.9648	0.9847	0.9412	0.9784	0.9878
Botnet	0.9348	0.9788	0.9897	0.9483	0.9828	0.9897

From Table 2, the accuracy for benign instances is 96.59%, 95.29%, and 91.47% with thresholds of 0.15, 0.1, and 0.05, respectively. Decreasing the threshold generally increases accuracy across all classes in both datasets. CSECICIDS2018 and CICIDS2017 consistently shows higher accuracy at all thresholds. The AE effectively identifies benign instances, with accuracy ranging from 91.47% to 96.59% in CICIDS2017 and 92.48% to 98.89% in CSECICIDS2018. Additionally, attacks such as denial-of-service (DoS), distributed DoS (DDoS), Brute Force, and Botnet are accurately detected, with accuracy consistently above 90% for most thresholds and datasets. Lower thresholds (0.05) tend to yield higher accuracy for most classes, indicating a preference for higher sensitivity at the expense of specificity. Overall, the AE exhibits robust performance in detecting both benign and attack instances across various thresholds and datasets, with generally higher accuracy observed in the both CICIDS2017 and CSECICIDS2018 dataset.

4.4. Assessing the OCSVM on the CICIDS2017 and CSECICIDS2018 datasets

The OCSVM is trained exclusively with non-malicious occurrences. To facilitate this training, a specific 'v' value was designated for the OCSVM. The 'v' parameter ranges between 0 and 1, defining both the minimum and maximum thresholds for the number of examples identified as support vectors, including those that lie on the wrong side of the hyperplane. The default value for 'v' is set to 0.5, implying that it

incorporates 50% of the training samples within the hyperplane. However, for the purpose of this experiment, several 'v' values were selected (0.2, 0.15, 0.1). These values were chosen to evaluate and gauge the performance of the AE. Seventy percent of the non-attack samples are utilized to train the OCSVM model. In contrast to the AE model, the evaluation of an OCSVM trained model results in a binary value 0 and 1. The output indicates whether an occurrence is classified within the SVM's fitted class. Consequently, the assessment of each attack relies on the number of instances predicted with a '0' output from the OCSVM.

Figure 6 presents a summary of the results obtained from the OCSVM. Upon examining the OCSVM results, two observations become apparent. Firstly, altering the 'v' value does not notably impact detection accuracy. Secondly, the classes that exhibit high accuracy in the AE results in Table 2 are similarly identified by the OCSVM. Nevertheless, it does not succeed in identifying two additional categories: those exhibiting enhanced accuracy at lower thresholds, and those demonstrating poor detection accuracy. This arises from the constraints of the OCSVM algorithm, which aims to establish a spherical hyperplane to distinguish the benign class from others. Nonetheless, classes falling within this hyperplane are consistently classified as benign. This can be further visualized in Figure 7. OCSVM is particularly effective in identifying recognizable zero-day attacks. Yet, AE excel in handling intricate zero-day attacks, boasting notably superior performance rankings.

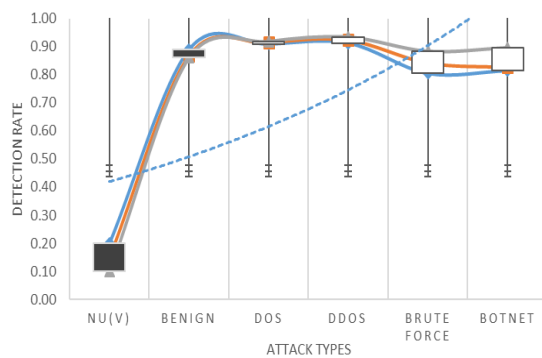


Figure 6. OCSVM accuracy evaluation on CICIDS2017

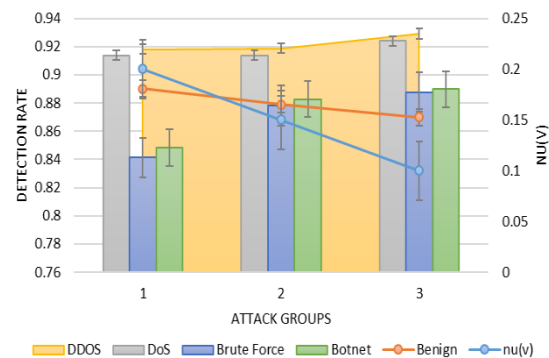


Figure 7. OCSVM accuracy evaluation on CSECICIDS2018

4.5. Performance comparison

Figure 8 illustrates AE and OCSVM algorithm performance across varied hyperparameters on CICIDS2017 and CSECICIDS2018 datasets. AE's detection accuracy improves with decreasing hyperparameters, consistently outperforming OCSVM. AE achieves highest accuracy on CICIDS2017 (98.88%) and CSECICIDS2018 (98.91%) with a hyperparameter of 0.05. OCSVM's accuracy also improves with reduced hyperparameters but remains slightly inferior to AE, notably on CICIDS2017. Its highest accuracies are 90.74% on CICIDS2017 and 90.76% on CSECICIDS2018 with a hyperparameter of 0.1. Overall, AE exhibits superior performance, with decreasing hyperparameters correlating with enhanced accuracy, suggesting CSECICIDS2018's potential for zero-day detection.

The radar chart in Figure 9 compares the accuracy of various anomaly detection algorithms for zero-day attack detection. AE leads with 98.90% accuracy, followed closely by OCSVM at 90.75%. K-nearest neighbors (KNN) [26] trails behind at 82.17%, indicating weaker performance compared to AE and OCSVM. Density-based spatial clustering of applications with noise (DBSCAN) [27] achieves 86.54% accuracy, falling between AE and OCSVM. Overall, AE demonstrates superior accuracy among the evaluated algorithms, with OCSVM, DBSCAN, and KNN following suit.

The combined results and discussion underscore the strong performance of our detection model, with the AE consistently achieving high accuracy on the CICIDS2017 and CSECICIDS2018 datasets. Key findings indicate that our model effectively detects both benign and attack instances, maintaining accuracies consistently above 90% across most thresholds. Figures 5 and 6 visually support these results, providing clear evidence of the model's effectiveness.

Our model demonstrates superior accuracy compared to previous studies, especially in identifying zero-day attacks. Although the AE outperforms the OCSVM, our study also highlights some limitations, such as the OCSVM's difficulty in detecting specific attack categories. Surprisingly, lower thresholds led to higher accuracy, indicating a trade-off between sensitivity and specificity.

The main objective of this study was to create an effective anomaly detection model. Our findings highlight the AE's crucial role in improving detection accuracy, which has important implications for future cybersecurity applications. Future research should focus on further optimizing hyperparameters and addressing the identified limitations, especially for more complex attack scenarios.

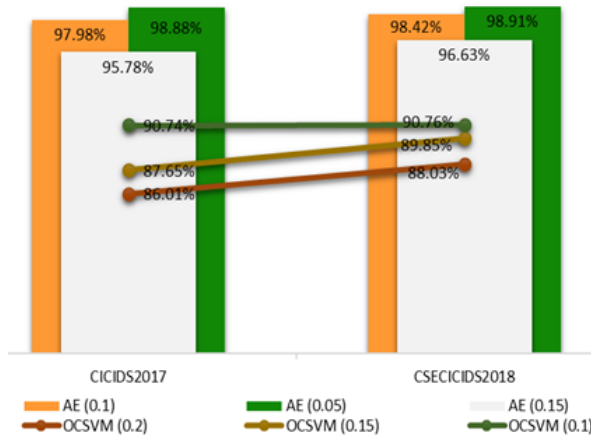


Figure 8. Performance comparison of AE with OCSVM

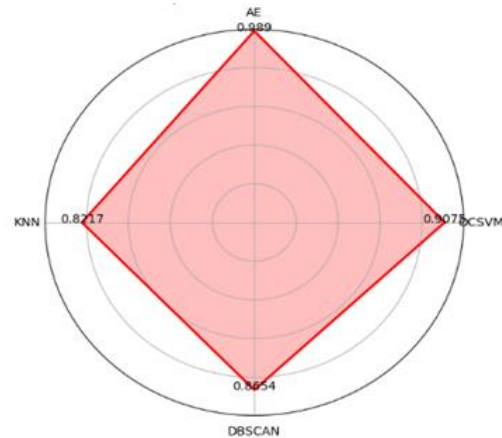


Figure 9. Performance comparison with Benchmark algorithms

5. CONCLUSION

The manuscript introduces an innovative approach to detect zero-day cyber-attacks using outlier-based techniques, aiming to address limitations of existing IDS. The proposed method revolves around an intelligent IDS model employing AE architecture, capitalizing on its encoding-decoding capabilities. Across the CICIDS2017 and CSEICIDS2018 datasets, the AE model exhibits robust detection accuracy, ranging from 98.47% to 98.99%. Comparative analysis against OCSVM underscores the superior accuracy of the proposed model, accompanied by low false-positive rates. Future endeavors entail assessing these models with specialized network IDS datasets, such as those pertaining to IoT and critical Infrastructure networks. This iterative process aims to adapt and refine the approach, fostering deeper insights into zero-day attack detection and facilitating exploration of additional ML techniques to enhance efficacy.




REFERENCES

- [1] M. Sarhan, S. Layeghy, M. Gallagher, and M. Portmann, "From zero-shot machine learning to zero-day attack detection," *International Journal of Information Security*, vol. 22, no. 4, pp. 947–959, Aug. 2023, doi: 10.1007/s10207-023-00676-0.
- [2] M. Paricherla, M. Ritonga, S. R. Shinde, S. M. Chaudhari, R. Linur, and A. Raghuvanshi, "Machine learning techniques for accurate classification and detection of intrusions in computer network," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 4, pp. 2340–2347, Aug. 2023, doi: 10.11591/eei.v12i4.4708.
- [3] I. Mbona and J. H. P. Eloff, "Detecting zero-day intrusion attacks using semi-supervised machine learning approaches," *IEEE Access*, vol. 10, pp. 69822–69838, 2022, doi: 10.1109/ACCESS.2022.3187116.
- [4] P. Chandre, P. Mahalle, and G. Shinde, "Intrusion prevention system using convolutional neural network for wireless sensor network," *IAES International Journal of Artificial Intelligence*, vol. 11, no. 2, pp. 504–515, Jun. 2022, doi: 10.11591/ijai.v11.i2.pp504-515.
- [5] A. S. Alfoudi *et al.*, "Hyper clustering model for dynamic network intrusion detection," *IET Communications*, 2022, doi: 10.1049/cmu2.12523.
- [6] B. M. Serinelli, A. Collen, and N. A. Nijdam, "On the analysis of open source datasets: Validating IDS implementation for well-known and zero day attack detection," in *Procedia Computer Science*, Elsevier B.V., 2021, pp. 192–199, doi: 10.1016/j.procs.2021.07.024.
- [7] S. Ali, S. U. Rehman, A. Imran, G. Adeem, Z. Iqbal, and K. Il Kim, "Comparative evaluation of AI-based techniques for zero-day attacks detection," *Electronics (Switzerland)*, vol. 11, no. 23, Dec. 2022, doi: 10.3390/electronics11233934.
- [8] S. Sakthimurugan, A. Sanjay Kumar, V. Vignesh, and P. Santhi, "Assessment of zero-day vulnerability using machine learning approach," *EAI Endorsed Transactions on Internet of Things*, vol. 10, 2024, doi: 10.4108/eetiot.4978.
- [9] T. Zoppi, A. Ceccarelli, and A. Bondavalli, "Unsupervised algorithms to detect zero-day attacks: strategy and application," *IEEE Access*, vol. 9, pp. 90603–90615, 2021, doi: 10.1109/ACCESS.2021.3090957.
- [10] M. Kante, V. Sharma, and K. Gupta, "Mitigating ransomware attacks through cyber threat intelligence and machine learning," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 33, no. 3, pp. 1958–1965, Mar. 2024, doi: 10.11591/ijeecs.v33.i3.pp1958-1965.
- [11] Y. Fang and G. Mogos, "Detecting attacks on e-mail," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 33, no. 3, pp. 1576–1588, Mar. 2024, doi: 10.11591/ijeecs.v33.i3.pp1576-1588.




- [12] H. A. A. Essa and W. S. Bhaya, "Ensemble learning classifiers hybrid feature selection for enhancing performance of intrusion detection system," *Bulletin of Electrical Engineering and Informatics*, vol. 13, no. 1, pp. 665–676, Feb. 2024, doi: 10.11591/eei.v13i1.5844.
- [13] B. I. Hairab, H. K. Aslan, M. S. Elsayed, A. D. Jurcut, and M. A. Azer, "Anomaly detection of zero-day attacks based on CNN and regularization techniques," *Electronics (Switzerland)*, vol. 12, no. 3, Feb. 2023, doi: 10.3390/electronics12030573.
- [14] X. Sun, J. Dai, P. Liu, A. Singhal, and J. Yen, "Using Bayesian networks for probabilistic identification of zero-day attack paths," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 10, pp. 2506–2521, 2018, doi: 10.1109/TIFS.2018.2821095.
- [15] N. H. Duong and H. D. Hai, "A semi-supervised model for network traffic anomaly detection," in *International Conference on Advanced Communication Technology, ICACT, 2015*, pp. 70–75, doi: 10.1109/ICACT.2015.7224759.
- [16] Y. Ren, K. Feng, F. Hu, L. Chen, and Y. Chen, "A Lightweight Unsupervised Intrusion Detection Model Based on Variational Auto-Encoder," *Sensors (Basel)*, vol. 23, no. 20, Oct. 2023, doi: 10.3390/s23208407.
- [17] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," *Biometrika*, 1986, [Online]. Available: <https://api.semanticscholar.org/CorpusID:62245742>
- [18] C. Brunner, A. Ko, and S. Fodor, "An autoencoder-enhanced stacking neural network model for increasing the performance of intrusion detection," *Journal of Artificial Intelligence and Soft Computing Research*, vol. 12, no. 2, pp. 149–163, Apr. 2021, doi: 10.2478/jaiscr-2022-0010.
- [19] W. Zhang and Y. Zhang, "Intrusion detection model for industrial internet of things based on improved autoencoder," *Computational Intelligence and Neuroscience*, vol. 2022, 2022, doi: 10.1155/2022/1406214.
- [20] M. Leon, T. Markovic, and S. Punnekkat, "Feature encoding with autoencoder and differential evolution for network intrusion detection using machine learning," in *GECCO 2022 Companion - Proceedings of the 2022 Genetic and Evolutionary Computation Conference*, Association for Computing Machinery, Inc, Jul. 2022, pp. 2152–2159. doi: 10.1145/3520304.3534009.
- [21] C. Wang, H. Liu, C. Li, Y. Sun, W. Wang, and B. Wang, "Robust intrusion detection for industrial control systems using improved autoencoder and Bayesian Gaussian mixture model," *Mathematics*, vol. 11, no. 9, May 2023, doi: 10.3390/math11092048.
- [22] A. Binbusayyis and T. Vaiyapuri, "Unsupervised deep learning approach for network intrusion detection combining convolutional autoencoder and one-class SVM," *Applied Intelligence*, vol. 51, no. 10, pp. 7094–7108, 2021, doi: 10.1007/s10489-021-02205-9.
- [23] C. Wang, Y. Sun, S. Lv, C. Wang, H. Liu, and B. Wang, "Intrusion detection system based on one-class support vector machine and Gaussian mixture model," *Electronics (Switzerland)*, vol. 12, no. 4, Feb. 2023, doi: 10.3390/electronics12040930.
- [24] M. A. Almaiah *et al.*, "Performance investigation of principal component analysis for intrusion detection system using different support vector machine kernels," *Electronics (Switzerland)*, vol. 11, no. 21, Nov. 2022, doi: 10.3390/electronics11213571.
- [25] S. Sokkalingam and R. Ramakrishnan, "An intelligent intrusion detection system for distributed denial of service attacks: A support vector machine with hybrid optimization algorithm based approach," *Concurrency and Computation*, vol. 34, no. 27, p. e7334, 2022, doi: 10.1002/cpe.7334.
- [26] Z. Liu, B. Xu, B. Cheng, X. Hu, and M. Darbandi, "Intrusion detection systems in the cloud computing: A comprehensive and deep literature review," *Concurrency and Computation*, vol. 34, no. 4, p. e6646, 2022, doi: 10.1002/cpe.6646.
- [27] Harintaka and C. Wijaya, "Automatic point cloud segmentation using RANSAC and DBSCAN algorithm for indoor model," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 21, no. 6, pp. 1317–1325, 2023, doi: 10.12928/TELKOMNIKA.V21i6.25299.

BIOGRAPHIES OF AUTHORS






Rajakumar Perumal    Assistant Professor in School of Computer Science and Engineering at Galgotias University, holds 22 years of teaching experience and is pursuing a Ph.D. in Computer Science and Engineering at Shri Venkateshwara University. With an M.E CSE from Anna University, he has published 4 patents and 6 research papers, specializing in networks, cloud computing, software engineering, and machine learning. He can be contacted at email: rajakumar.jcet@gmail.com.






Dr. Tamilarasi Karuppiah    Associate Professor in Department of Information Technology at Panimalar Engineering College, accumulating 24 years of teaching experience. She earned her Ph.D. record with 7 patents, 5 book chapters, and 18 research papers published in esteemed international journals and conferences. Her expertise spans cyber security, networks, cloud computing, and machine learning. She can be contacted at email: thamizhanna@gmail.com.






Uppiliraja Panneerselvam    Assistant Professor in School of Computer Science and Engineering at Galgotias University, holds 11 years of teaching experience. With an M.E CSE from Anna University, he has published 2 patents and 5 research papers, specializing in cloud computing, computer networks, software engineering, and machine learning. He can be contacted at email: uppiraja@gmail.com.



Venkatesan Annamalai    Assistant Professor in Department of Electrical, Electronics and Communication Engineering at Galgotias University, holds 14 years of teaching experience. With an M.E Comm. Systems from Anna University, he has published 3 patents and 4 research papers, specializing in computer networks, telecommunications, cybersecurity in communication systems and machine learning. He can be contacted at email: venkatesan@galgotiasuniversity.edu.in.



Prabu Kaliyaperumal    Assistant Professor in School of Computer Science and Engineering at Galgotias University, has 16 years of teaching experience. Currently pursuing a Ph.D, he holds an M.Tech in CSE from SRM University and MBA from Anna University. He has published 4 patents and 9 research papers in international journals and conferences. His expertise includes cyber security, networks, cloud computing, and machine learning. He can be contacted at mega.prabu@gmail.com.