International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 8s
DOI: https://doi.org/10.17762/ijritcc.v11i8s.7190
***Article Received**: 22 April 2023 **Revised:** 10 June 2023 **Accepted:** 26 June 2023*

Zero-day Network Intrusion Detection using Machine Learning Approach

Naushad Alam[1], Muqeem Ahmed[2]
[1]Department of Computer Science & Information Technology
Maulana Azad National Urdu University
Hyderabad-500032
Email- alamnaushad290@gmail.com
[2]Department of Computer Science & Information Technology
Maulana Azad National Urdu University
Hyderabad-500032
Email- muqeem.ahmed@gmail.com

Abstract-Zero-day network attacks are a growing global cybersecurity concern. Hackers exploit vulnerabilities in network systems, making network traffic analysis crucial in detecting and mitigating unauthorized attacks. However, inadequate and ineffective network traffic analysis can lead to prolonged network compromises. To address this, machine learning-based zero-day network intrusion detection systems (ZDNIDS) rely on monitoring and collecting relevant information from network traffic data. The selection of pertinent features is essential for optimal ZDNIDS performance given the voluminous nature of network traffic data, characterized by attributes. Unfortunately, current machine learning models utilized in this field exhibit inefficiency in detecting zero-day network attacks, resulting in a high false alarm rate and overall performance degradation. To overcome these limitations, this paper introduces a novel approach combining the anomaly-based extended isolation forest algorithm with the BAT algorithm and Nevergrad. Furthermore, the proposed model was evaluated using 5G network traffic, showcasing its effectiveness in efficiently detecting both known and unknown attacks, thereby reducing false alarms when compared to existing systems. This advancement contributes to improved internet security.

Keywords: Cybersecurity; Zero-day attack; BAT algorithm; Nevergrad.

I. INTRODUCTION

In the technological era we live in, the Internet has become a necessary tool for business, education, and entertainment. The Internet has become a vital part of our day-to-day routines. Today, it is regarded as one of the most critical elements of the modern business landscape [1]. Network usage is on the rise, which carries with it the risk of attack. Keeping systems and networks secure is getting harder every year. Protecting against threats in real time is a challenging endeavor, and one of the most important aspects of cyber defense is reducing false alarm rates. According to the report [2], the number of vulnerabilities discovered each year has been growing continuously, with a total of 233,758 new vulnerabilities discovered in 2022 alone. The number of exploits has also been continually increasing, with 18.3 million discovered in 2022, in which phishing attack variants had the highest occurrence (41%), followed by malware and ransomware attacks (26%). The data suggests that zero days have been increasingly popular among attackers in recent years, with 192 zero days found in 2022. This is a huge increase over previous years.

Cybersecurity safeguards against attacks, data loss, and unauthorized access for internet-connected devices such as networks, computers, apps, and computers. [3]. The intrusion detection system (IDS) is a crucial component of cybersecurity systems. Its purpose is to detect, analyze, and identify unauthorized intrusions by examining data collected from network devices. [4]. There has been a significant increase in research on network intrusion detection over the past few years, and there are numerous opportunities to advance the state-of-the-art in detecting and preventing network-based attacks, despite significant progress and a substantial corpus of work [5]. IDSs are programs that continuously monitor computer networks for harmful activities. Unauthorized attempts to steal sensitive information, censorship of network protocols, or any other breach of network security protocols are examples of such operations. IDSs provide an additional layer of protection to help avoid successful network attacks by detecting such actions and generating alerts. IDS use two types of detection mechanisms: signature-based and anomaly-based detection [6]. Signature-based intrusion detection uses a database of known attack patterns to detect intrusion, which is effective but only for known attacks. However, it cannot protect against unknown attacks or "zero-day attacks" that are not in the database. Updating the database or server is very time-consuming and unfeasible**.** Anomaly detection approaches build normal-operation profiles using the system's regular activity and identify anomalies as any behavior that deviates from the norm.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 8s
DOI: https://doi.org/10.17762/ijritcc.v11i8s.7190
***Article Received**: 22 April 2023 **Revised:** 10 June 2023 **Accepted:** 26 June 2023*

They have the potential to detect all types of attacks, known or unknown, including zero-day attacks. The fundamental problem with anomaly-based detection techniques is that they need a tuning stage and have high false-positive rates. Many studies suggest using machine learning techniques for cyber intrusion detection to enhance the detection rate and reduce false-positive rates [3, 6, 7]. Recent research focuses on anomaly-based intrusion detection systems, which can be classified into three types based on the machine learning methods used: supervised (classification), unsupervised (clustering and anomaly-based detection), reinforcement and semi-supervised techniques [6]. Supervised IDS uses labeled data to train a model [8, 9]. However, when compared to signature-based IDS, supervised IDS models are less efficient at detecting zero-day attacks, and they require frequent retraining, which is difficult to achieve because obtaining labeled data is difficult. Semi-supervised IDS utilizes a combination of labeled and unlabeled data to build a model. In unsupervised IDS, clustering algorithms are employed to identify anomalies in unlabeled data. These techniques aim to group similar data together while maintaining dissimilarity between clusters without relying on attack signatures, explicit attack descriptions, or labeled data for training. Unsupervised intrusion detection methods have the capability to detect both known and unknown attacks, eliminating the requirement for labeled data. These methods can extract features from different sources to address queries related to attribution and correlation [6].

In this paper, we introduce an unsupervised anomaly detection method that does not rely on prior knowledge. Our approach offers three key contributions. Firstly, we present a unique anomaly detection method that combines extended isolation forest with the BAT algorithm. Secondly, we optimize the extended isolation forest using Nevergrad. Lastly, we conduct experiments to compare our proposed method with three alternative approaches, utilizing diverse evaluation metrics.

Nevergrad is a Python library that offers a gradient-free optimization platform [39]. Its purpose is to optimize complex functions and models without relying on gradients, making it a valuable tool for machine learning and optimization endeavors. Users can leverage Nevergrad to minimize objective functions, fine-tune hyperparameters, and execute various optimization tasks efficiently. The scalability of Nevergrad enables its application across diverse domains, facilitating optimization for a wide range of applications.

II. RELATED WORK

Numerous machine learning and data mining techniques have been suggested for cyber intrusion detection in the past twenty years. These include ant colony optimization [10], artificial neural networks [11, 12], particle swarm optimization [13], evolutionary computation [14], Support Vector Machine (SVM) [15], and Benford's law with semi-supervised machine learning [16]. The "deep transductive transfer learning" method proposed in this research can identify zero-day assaults even in the absence of labeled data in the target domain. The outcomes of the experiments demonstrate how well this approach can spot zero-day assaults on fresh data. In plain language, the suggested method can identify previously unidentified cyberattacks without the requirement for prior information [17]. Paper [18], In their work, they make the recommendation that autoencoders be used to create an IDS model that is capable of accurately detecting zero-day attacks with a high recall and low false-negative rate. According to the study, autoencoders are effective at spotting sophisticated zero-day attacks. The report also emphasizes how the suggested technique trades off fallout and memory. In this paper [19], a deep learning-based method for creating a nimble and effective network intrusion detection system (NIDS) is presented. In terms of accuracy, precision, recall, and f-measure values, the suggested method's performance is assessed and contrasted to earlier methods. The final objective is to use deep learning methods to construct a real-time NIDS for genuine networks. The author [20] is to conduct a thorough examination of the NSL-KDD dataset by extracting pertinent records and comparing different machine learning classifiers. The trials' findings demonstrated that of all the evaluated models, the Random Forest classifier had the highest average accuracy and outperformed them in numerous tests. The performance of various classifiers on the NSL-KDD dataset is discussed in this work. This study's goal is to create an intrusion detection system with high detection rates and low false alarm rates. According to the experimental findings, the feature association impact scale (FAIS) model with all characteristics had an accuracy of 88%, whereas the feature correlation analysis and association impact scale (FCAAIS) model with optimal features had an accuracy of 91%. The accuracy of FAIS was increased by 3% with the use of canonical correlation for optimized attribute selection. Calculations of sensitivity, specificity, and F-measure revealed FCAAIS to have greater values than FAIS [21]. The key idea behind our investigation is to identify executable files connected to known vulnerabilities and their exploits. These discoveries have important ramifications for both upcoming security technology and governmental initiatives. We can strengthen security protocols and provide more robust mitigation solutions for possible security risks by recognizing such files. The study's conclusions can influence current and upcoming cybersecurity research and development projects [22]. In order to evaluate how well machine learning-based NIDSs are able to identify zero-day attacks, this study introduces a unique zero-shot learning technique. Despite strong zero-day detection rate (Z-DR) values in the majority of

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 8s
DOI: https://doi.org/10.17762/ijritcc.v11i8s.7190
***Article Received**: 22 April 2023 **Revised:** 10 June 2023 **Accepted:** 26 June 2023*

attack classes, the study's findings show that some attack categories were not reliably recognized as zero-day threats. The Wasserstein Distance (WD) method, which directly connected feature distributions with WD and Z-DR measures, was used to further corroborate the findings [23]. In this study [24], 356 severe attacks employing an out-of-date official rule set were used to test Snort's capacity to recognize zero-day attacks. The analysis' findings demonstrated that Snort has a 17% detection rate for zero-day attacks. The reputation architecture for vehicular ad hoc networks, Clustered, is presented in this study. It involves the cluster chiefs and members altering pseudonyms and reputation values. Studies reveal that it is more scalable and efficient than competing approaches, but there is still room for improvement in terms of flexibility and resilience in the dynamic vehicular ad hoc network (VANET) environment [25]. A proactive network security strategy is presented that makes use of deep learning models to identify intrusions. The machine learning application's development and deployment are both covered by the suggested system architecture. The system can produce high-quality model performance in dynamic and quickly changing situations by fusing deep learning modeling with scalable data pre-processing [26]. This paper [27] presents an add-on for IoT devices that detects URL-based attack using a convolutional neural network (CNN) model and botnet attacks using recurrent neural network-long short term memory model housed on back-end servers. The add-on is intended to improve IoT device micro security. A bidirectional long and short-term memory network with multi feature layer for successful attack detection with various intervals is proposed. In comparison to prior methods, the model's introduction of sequence and stage feature layers and a double-layer reverse unit results in a decreased false positive and false negative rate [28]. A novel deep learning approach for intrusion detection that outperforms previous approaches in terms of accuracy, precision, and recall while requiring less training time The method was tested on the KDD Cup '99 and NSL-KDD datasets, showing an improvement in accuracy of up to 5%. GPUs were used to build the classifier in TensorFlow [29].

Table 1: Brief Literature Survey of The Related Work

Model	Methods	Accuracy	Gap/ Future work
CNN-SVM [30]	Signature based Of Double-layered Hybrid approach	R2L-96.67, U2R-100	Future researchers can test the categorization's efficacy by using it on a dataset or network setting with more than four different types of attacks.
CNN- DCNN-LSTM [27]	Deep Learning	CNN - 94.3, F1-93.58	The suggested method could be enhanced in the future to recognize new attacks on IoT systems and devices that use encrypted traffic to evade detection or hide their activities.
Machine Learning Model for NIDS [31]	Semi Supervised Machine Learning	Correlation coefficient-74 & F1-score-85	a system that combines several feature selection strategies with Machine Learning classifiers for improved performance requirements additional research.
Deep Learning [19]	NDAE for unsupervised learning	98.81	In upcoming research, the researcher aims to enhance the model's ability to detect zero-day attacks and evaluate it further using world -backbone network traffic.
Unsupervised learning [32]	Deep learning based unsupervised learning algorithm: K-Mean, SOM, DAGMM and ALAD	K-Mean -97.6 ALAD – 89.9 SOM -96.1	In this paper, K-Mean and SOM are reliable, but ALAD is better at detecting rare attacks by using adversarial samples, and DAGMM doesn't perform well. Test the algorithm on additional dataset and combine for network flow anomaly detection.

III. DATASET DESCRIPTION

The 5G-NIDD dataset [33] is a comprehensive labeled dataset generated from a functional 5G test network. Its purpose is to facilitate the identification and detection of malicious content within network traffic. This dataset comprises substantial amounts of data collected from actual networks. A recent survey conducted shows a brief overview of the datasets available till 2020 that are useful for evaluating intrusion detection on networks [34]. Many of the existing datasets are outdated and may not be suitable for analyzing modern networks due to significant technological advancements. However, the 5G-NIDD [33] dataset is a recent compilation that incorporates real 5G networks. It encompasses prevalent attacks, such as different port scans and a diverse range of DoS/DDoS attacks.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 8s
DOI: https://doi.org/10.17762/ijritcc.v11i8s.7190
***Article Received**: 22 April 2023 **Revised:** 10 June 2023 **Accepted:** 26 June 2023*

IV. METHODOLOGY

This research consists of two main parts. The first part is feature selection using a metaheuristic algorithm BAT by training the Extended Isolation Forest (EIF). The second part is training the model using selected features and tuning the hyperparameters with the Nevergrad optimizer for evaluating the new methodology. Further comparing it with other methods to determine its effectiveness. figure 1 show the overview of our methodology.

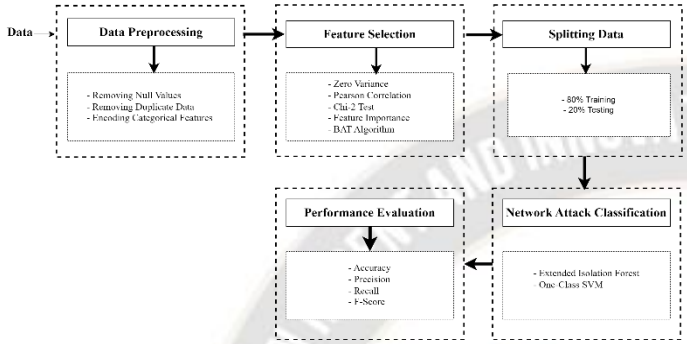

Fig 1: Overview of purposed methodology

This study aims to select important features from network packets using BAT-based optimization as a wrapper classifier. The selected subset of features is the output of the BATEIF algorithm, which improves the detection capacity of the system. Further, we use Nevergrad to optimize hyperparameters for the best results. The presented system is then tested and evaluated for its effectiveness in increasing detection accuracy and reducing false alarms. The 5G NIDD dataset is used as a benchmark to assess the system's performance.

BAT is a metaheuristic optimization algorithm that draws inspiration from how bats use echolocation to find prey. The key aspects of this behavior are simplified as follows [35]:

- Bats use echolocation to detect the distance to obstacles and prey.
- Bats fly at random using a velocity v_i and emit pulses with a frequency f_{min}, wavelength λ and loudness A_0 to find prey. Bats are able to adjust the frequency and rate of their echolocation pulses based on how close they are to obstacles in their environment. This adjustment happens spontaneously. The pulse emission rate is a value that ranges from 0 to 1.
- The loudness of the bat's echolocation varies from a high value of $A_0$to a minimal value A_{min}.

BATEIF ALGORITHM

1. Initialization: Randomly initialize a population of n_bats, each represented by a binary vector of length N. Also initializes the velocity v of each bat to zero and sets the initial fitness values to zero.
2. Frequency and velocity update:
   f[j] = A * exp(-ri) * cos(2pi*rand()) + gamma
   v[j] += (bats[j] - bats.mean(axis=0)) * f[j]
3. Position update:
   bats += v
4. Loudness and pulse rate update:
   loudness = alpha * loudness
   pulse_rate = exp(-gamma*i)
5. Fitness evaluation: The fitness of each bat is evaluated using a fitness function that measures the accuracy of an EIF trained on the subset of features selected by the binary vector representation of the bat.
6. Update of best solution: If a bat's fitness value is better than the current best fitness value, the bat's binary vector representation is set as the new best solution.
7. Termination: The algorithm terminates after a fixed number of iterations, and the final solution is the binary vector representation of the best bat.
8. The final step is to select the features corresponding to the binary vector representation of the best bat and return them as the selected features.

Here, N is the number of features in the input data. In each iteration of the algorithm, the frequency f and velocity v of each bat are updated according to the above equations. After that, i is the current iteration number, j is the index of the current bat, A is a constant representing the initial loudness of the bat's calls, r and gamma are constants controlling the decay rates of loudness and frequency, and rand() generates a random number between 0 and 1. The position of each bat is then updated by adding its velocity vector to its current position. In the next step, the loudness and pulse rate of each bat are updated according to the equations. Alpha is a constant that controls the rate of loudness decay.

The BAT algorithm has some advantages that make it a useful tool for solving classification and time series prediction problems. Here are a few of these advantages [20]. First, the BAT uses echolocation and frequency tuning to

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 8s
DOI: https://doi.org/10.17762/ijritcc.v11i8s.7190
***Article Received**: 22 April 2023 **Revised:** 10 June 2023 **Accepted:** 26 June 2023*

adjust its behavior during the problem-solving process. The second allows it to adjust the frequency of its pulses to fine-tune its search. The BAT can automatically zoom in on areas where potentially better solutions might be found. The BAT algorithm possesses the advantage of rapidly converging on optimal solutions during the initial stages of the iteration process. Unlike several other algorithms, the BAT algorithm incorporates parameter control, enabling automatic adjustment of parameter values (A and r) throughout the iterations. This adaptive capability facilitates a seamless transition from exploration to exploitation, enhancing the algorithm's effectiveness in searching for the best solution.

V. DATA PREPROCESSING AND FEATURE SELECTION

Machine learning intrusion detection systems (ML-NIDS) use input data called features to detect zero-day network attacks [36]. ML-NIDS (Machine Learning-based Network Intrusion Detection Systems) can enhance their performance by leveraging crucial features that differentiate normal and anomalous network traffic.

Network traffic analysis (NTA) is a crucial component of Network Intrusion Detection Systems (NIDS) that involves capturing and analyzing network traffic data. Its primary objective is to identify and detect various threats, including zero-day network attacks. Nonetheless, the real-time extraction of significant features from network traffic data presents a challenge. Significance is attributed to a network traffic feature if it demonstrates the capability to distinguish between normal and malicious traffic. The information regarding significant features is sourced from references [37, 38]. Our study aimed to tackle the issue of effectively extracting significant features to detect unfamiliar malicious attacks. In our approach, we first remove the duplicate data, dropping some columns due to the maximum data present in columns being zero and categorical features available in the dataset. So, encode these columns using one hot encoding. Secondly, we are checking the zero variance and Pearson correlation on all features and dropping redundant features because these features reduce the accuracy of our model. After that, we applied the chi-2 test to feature importance to check the contribution of every feature. Further, we have applied BATEIF to identify the most important features (selected features by BATEIF) from the dataset by searching for a subset of features that maximizes the performance of the model. How well an ZDNIDS performs in terms of precision, recall, and F1 score determines the efficiency of the features chosen. The next section discusses the implementation of various ML models (EIF and OneClass SVM) for detecting zero-day attacks.

VI. RESULT AND DISCUSSION

This section outlines the implementation of the Extended Isolation Forest (EIF) and OneClass SVM models for the detection of zero-day network attacks. Prior to training, the dataset is divided into two portions: a training dataset and a testing dataset, utilizing an 80/20 ratio. After that, we used all the features selected by BATEIF to train our model as well as tune hyperparameters using Nevergrad. Further, we used a well-known evaluation matrix named precision, F1 score, recall, and accuracy.

Accuracy = (Number of correctly classified instances) / (Total number of instances)

Precision = TP/TP+FP

Recall = TP/TP+FN

F1 score = 2*Precision*Recall/Precision+Recall

Fig. 2 shows a bar graph of two machine learning models: the EIF model with and without BATEIF. The first model (using BATEIF) has an accuracy of 99%. The second model (without BATEIF) has an accuracy of 58%. In this case, the first model is better than the second model. This is because the first model is more accurate and has a higher detection rate.

Fig. 3 shows a bar graph of two machine learning models: the OneClass SVM model with and without BATEIF. The first model (using BATEIF) has an accuracy of 90% where as 62% accuracy is achieved by the second model (without BATEIF). In this case, the first model is better than the second model. This is because the first model is more accurate and has a higher detection rate.

Fig. 4 shows A bar graph of two machine learning models has been shown: the EIF and the OneClass SVM model with BATEIF. The first model has 99% accuracy where as the accuracy of the second model is 90%. In this situation, the first model (EIF) outperforms the second model. This is due to the fact that the first model is more accurate and has a greater detection rate.

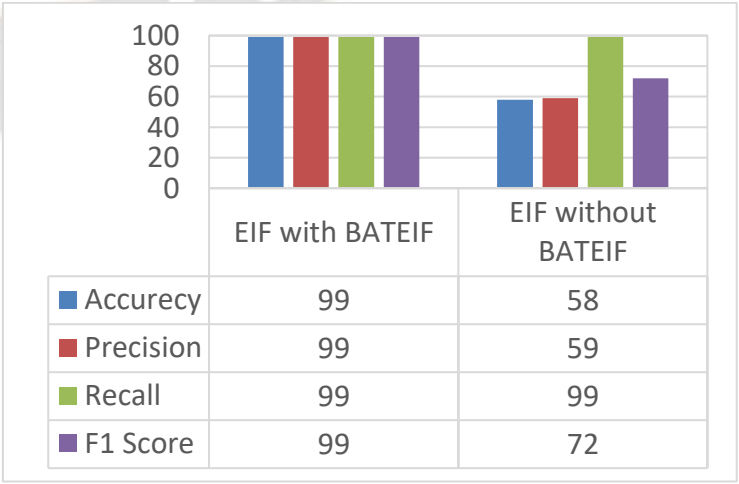

Fig 2: Comparative analysis between EIF model with and without BATEIF

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 8s
DOI: https://doi.org/10.17762/ijritcc.v11i8s.7190
***Article Received**: 22 April 2023 **Revised:** 10 June 2023 **Accepted:** 26 June 2023*

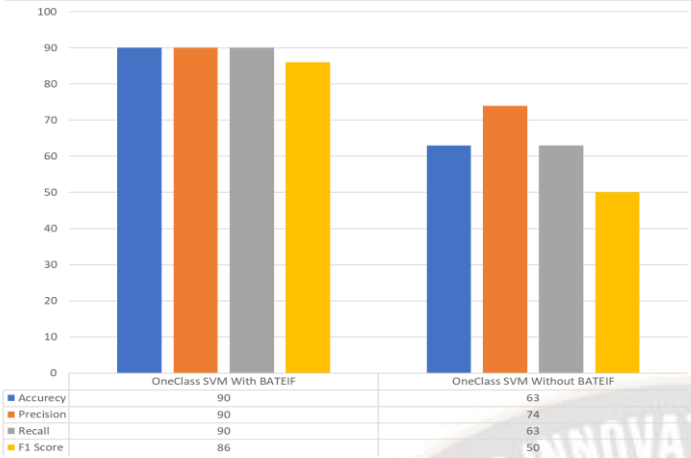

Fig 3: Comparative analysis between OneClass SVM model with and without BATEIF

Table 2: Selected features when training/testing against the 5G NIDD dataset

Dataset Name	Selected Feature Index	Total selected Feature
5G NIDD	1,3,6,9,10,16,25,26,28,29,31,38,40,42,47,52,56,57,58,60, 61,62, 64,65	24

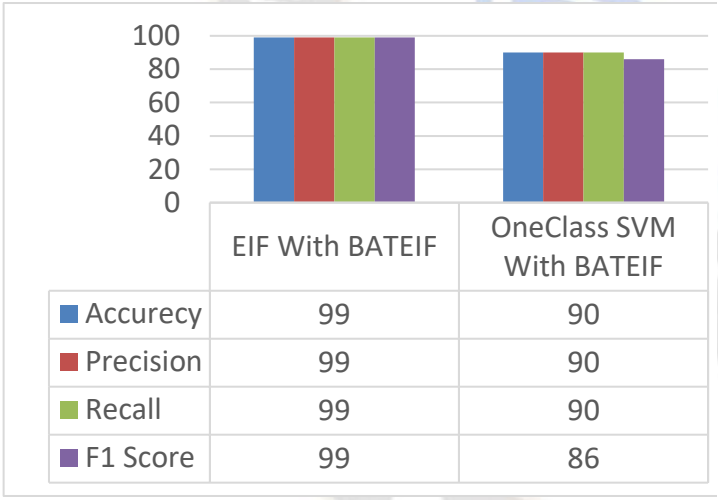

Fig 4: Comparative analysis between OneClass SVM model and EIF with BATEIF

In general, the network features identified by BATEIF have demonstrated superior performance compared to existing systems. However, certain features, such as the source port, destination port, and seq are not effective in detecting zero-day network attacks and consequently have a negative impact on the performance of our model.

The BATEIF algorithm with the Nevergrad optimizer is a novel algorithm for intrusion detection that has been shown to be effective at detecting both known and unknown attacks. The algorithm works by first identifying anomalous behavior in network traffic. This anomalous behavior is then used to build a model of the attack. The model is then used to detect new attacks that are similar to the known attacks. The extended isolation forest is a machine learning algorithm that has been shown to be effective at detecting outliers. These data points are significantly different from the rest of the data. The extended isolation forest works by first building a forest of decision trees. Each decision tree is used to classify data points as either normal or anomalous. The BAT algorithm with the extended isolation forest was evaluated on a 5G NIDD that contained both known and unknown attacks. We compared the performance of EIF and OneClass SVM using selected features by BATEIF and without. The results showed that the EIF, with the Nevergrad and BAT algorithms, is able to detect 99% of the known and unknown attacks. The results of this study show the BAT algorithm and the extended isolation forest combined are effective at detecting intrusions. The BAT algorithm is more effective for feature selection, while the extended isolation forest is more effective at detecting attacks. On the other hand, OneClass SVM takes too much time to train, although EIF does not.

VII. CONCLUSION

This research aims to present an effective and better way to detect the zero-day attacks. In this approach, firstly, BATEIF, a novel metaheuristic algorithm based on the binary version of the BAT algorithm, is presented for feature selection purposes. The first goal was used as a criterion to evaluate various methods for improving the quality of selected features: the number of features, the false-positive rate, and the rate of detection. The goal of choosing an excellent characteristic subset to train EIF and OneClass SVM that conduct intrusion detection. Finally, we tested on the most recent 5G NIDD dataset, and the results were great. F1 score with 99% accuracy, precision, recall, and recall. Attackers are always developing new and complicated techniques to attack weaknesses, making it difficult for IDS to keep up. The great results show the study's contributions to providing a better IDS.

REFERENCES

[1] T. Shon and J. Moon, “A hybrid machine learning approach to network anomaly detection,” Inf. Sci., vol. 177, no. 18, pp. 3799-3821, Sept. 2007, doi:10.1016/j.ins.2007.03.025.

[2] IBM, “Security X-force threat intelligence index 2023”. Available at: https://www.ibm.com/downloads/cas/DB4GL8YM.

[3] A. L. Buczak and E. Guven, “A survey of data mining and machine learning methods for cyber security intrusion detection,” IEEE Commun. Surv. Tutor., vol. 18, no. 2, pp. 1153-1176, 2016, doi:10.1109/COMST.2015.2494502.

[4] A. Mukkamala et al., “Cyber security challenges: Designing efficient intrusion detection systems and antivirus tools” in Proc. of Enhancing Computer Security with Smart Technology, New York, NY, USA, 2005, pp. 125-163.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 8s
DOI: https://doi.org/10.17762/ijritcc.v11i8s.7190
***Article Received**: 22 April 2023 **Revised:** 10 June 2023 **Accepted:** 26 June 2023*

[5] A. Sundaram, 'An introduction to intrusion detection,' Crossroads, vol. 2, no. 4, Apr. 1996, pp. 3-7.

[6] A. Nisioti et al., "From intrusion detection to attacker attribution: A comprehensive survey of unsupervised methods," IEEE Commun. Surv. Tutor., vol. 20, no. 4, pp. 3369-3388, 2018, doi:10.1109/COMST.2018.2854724.

[7] P. Casas et al., "Unsupervised network intrusion detection systems: Detecting the unknown without knowledge," Comput. Commun., vol. 35, no. 7, pp. 772-783, 2012, doi:10.1016/j.comcom.2012.01.016.

[8] I. Kang et al., "A differentiated oneclass classification method with applications to intrusion detection," Expert Syst. Appl., vol. 39, no. 4, pp. 3899-3905, 2012, doi:10.1016/j.eswa.2011.06.033.

[9] F. Kuang et al., "A novel hybrid KPCA and SVM with GA model for intrusion detection," Appl. Soft Comput., vol. 18, pp. 178-184, 2014, doi:10.1016/j.asoc.2014.01.028.

[10] L. Wang and J. Shen, "A systematic review of bio-inspired service concretization," IEEE Trans. Serv. Comput., vol. 10, no. 4, pp. 493-505, 2017, doi:10.1109/TSC.2015.2501300.

[11] L. Wang et al., "Feed-back neural networks with discrete weights," Neural Comput. Appl., vol. 22, no. 6, pp. 1063-1069, 2013, doi:10.1007/s00521-012-0867-8.

[12] H. Hindy, et al., "Utilising deep LearningTechniques for effective zero-day attack detection," Electronics, vol. 9, no. 10, p. 1684, 2020, doi:10.3390/electronics9101684.

[13] L. Wang and J. Shen, "Data-intensive service provision based on particle swarm optimization," Int. J. Comp. Intell. Syst., vol. 11, no. 1, pp. 330-339, 2018, doi:10.2991/ijcis.11.1.25.

[14] M. Sadiq and A. Khan, "Rule-based network intrusion detection using genetic algorithms," Int. J. Comput. Appl., vol. 18, no. 8, pp. 26-29, 2011, doi:10.5120/2303-2914.

[15] C. Wagner et al., "Machine learning approach for IP-flow record anomaly detection," Lect. Notes Comput. Sci., vol. 6640, pp. 28-39, 2011, doi:10.1007/978-3-642-20757-0_3.

[16] I. Mbona and J. H. P. Eloff, "Detecting zero-day intrusion attacks using semi-supervised machine learning approaches,", IEEE Access, vol. 10, 69822-69838, doi:10.1109/ACCESS.2022.3187116.

[17] N. Sameera and M. Shashi, "Deep transductive transfer learning framework for zero-day attack detection," ICT Express, vol. 6, no. 4, pp. 361-367, 2020, doi:10.1016/j.icte.2020.03.003.

[18] H. Hindy et al., "'Utilising deep learning techniques for effective zero-day attack detection,' Electron.,", Electronics, vol. 9, no. 10, pp. 1-16, 2020, doi:10.3390/electronics9101684.

[19] N. Altwaijry et al., "A Deep Learning Approach for Anomaly-Based Network Intrusion Detection" Commun. Comput. Inf. Sci., vol. 1210 CCIS, pp. 603-615, 2020, doi:10.1007/978-981-15-7530-3_46.

[20] D. P. Gaikwad and R. C. Thool, "'Online Anomaly Based Intrusion Detection System Using Machine Learning,' i-manager's J. Cloud Comput,", JCC, vol. 1, no. 1, pp. 19-25, 2014, doi:10.26634/jcc.1.1.2800.

[21] D. Oladimeji, "'an Intrusion Detection System for Internet of,' no," Jun., pp. 1-25, 2021.

[22] L. Bilge and T. Dumitraş, Before We Knew It: An Empirical Study of Zero-Day Attacks in the Real World, Proc. ACM Conf. Comput. Commun, 2012, pp. 833-844, doi:10.1145/2382196.2382284.

[23] Thomas Wilson, Andrew Evans, Alejandro Perez, Luis Pérez, Juan Martinez. Machine Learning for Anomaly Detection and Outlier Analysis in Decision Science. Kuwait Journal of Machine Learning, 2(3). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/207

[24] M. Sarhan et al., "From zero-shot machine learning to zero-day attack detection," Int. J. Inf. Secur., vol. 2023, no. Jun., 2019, doi:10.1007/s10207-023-00676-0.

[25] H. Holm, "Signature based intrusion detection for zero-day attacks: (Not) A closed chapter?," Proc. Annu. Hawaii Int. Conf., Syst. Sci., pp. 4895-4904, 2014, doi:10.1109/HICSS.2014.600.

[26] J. Wang et al., "ClusterRep: A cluster-based reputation framework for balancing privacy and trust in vehicular participatory sensing," Int. J. Distrib. Sens. Netw., vol. 14, no. 9, 2018, doi:10.1177/1550147718803299.

[27] G. Nguyen et al., "Deep learning for proactive network monitoring and security protection," IEEE Access, vol. 8, pp. 19696-19716, 2020, doi:10.1109/ACCESS.2020.2968718.

[28] G. De La Torre Parra et al., "Detecting Internet of Things attacks using distributed deep learning," J. Netw. Comput. Appl., vol. 163, no. Oct., 2020, doi:10.1016/j.jnca.2020.102662.

[29] X. Li et al., "Detection of low-frequency and multi-stage attacks in industrial Internet of things," IEEE Trans. Veh. Technol., vol. 69, no. 8, pp. 8820-8831, 2020, doi:10.1109/TVT.2020.2995133.

[30] S. Moraboena et al., "A deep learning approach to network intrusion detection using deep autoencoder," Rev. Intell. Artif., vol. 34, no. 4, pp. 457-463, 2020, doi:10.18280/ria.340410.

[31] T. Wisanwanichthan and M. Thammawichai, "A double-layered hybrid approach for network intrusion detection system using combined naive Bayes and SVM," IEEE Access, vol. 9, pp. 138432-138450, 2021, doi:10.1109/ACCESS.2021.3118573.

[32] I. Mbona and J. H. P. Eloff, "Detecting zero-day intrusion attacks using semi-supervised machine learning approaches," IEEE Access, vol. 10, no. Apr., pp. 69822-69838, 2022, doi:10.1109/ACCESS.2022.3187116.

[33] M. A. Kabir and X. Luo, "Unsupervised learning for network flow based anomaly detection in the era of deep learning," Proc 6th Int. Conf. Big Data Comput. Serv. Appl. Big Data Service 2020, vol. August 2020. IEEE, 2020, pp. 165-168, doi:10.1109/BigDataService49289.2020.00032.

[34] S. Samarakoon et al., 2022, 5G-NIDD: A comprehensive network intrusion detection dataset generated over 5G wireless network. arXiv preprint arXiv:2212.01298.

[35] K. Shaukat et al., "A survey on machine learning techniques for cyber security in the last decade," IEEE Access, vol. 8, pp. 222310-222354, 2020, doi:10.1109/ACCESS.2020.3041951, P. 222.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 8s
DOI: https://doi.org/10.17762/ijritcc.v11i8s.7190
***Article Received**: 22 April 2023 **Revised:** 10 June 2023 **Accepted:** 26 June 2023*

[36] X.-S. Yang, "A new metaheuristic bat-inspired algorithm" in Nature Inspired Cooperative Strategies for Optimization (NICSO 2010). Berlin, Germany: Springer, 2010, pp. 65-74, doi:10.1007/978-3-642-12538-6_6.

[37] R. Abdulhammed et al., "Features dimensionality reduction approaches for machine learning based network intrusion detection," Electronics, vol. 8, no. 3, p. 322, Mar. 2019, doi:10.3390/electronics8030322.

[38] E. Druică et al., "Benford's law and the limits of digit analysis," Int. J. Acc. Inf. Syst., vol. 31, pp. 75-82, Dec. 2018, doi:10.1016/j.accinf.2018.09.004.

[39] M. F. Umer et al., "Flow-based intrusion detection: Techniques and challenges," Comput. Secur., vol. 70, pp. 238-254, Sept. 2017, doi:10.1016/j.cose.2017.05.009.

[40] G. Biau et al., "Nevergrad – A gradient-free optimization platform," J. Mach. Learn. Res., vol. 21, no. 34, pp. 1-6, 2020.