2022 International Conference for Advancement in Technology (ICONAT)
Goa, India. Jan 21-22, 2022

An Intrusion Detection System for Zero-Day Attacks to Reduce False Positive Rates

Priya Pitre
Dept of Computer Engg and IT
College of Engineering, Pune (COEP)
Pune, India
pitrepn18.comp@coep.ac.in

Arya Gandhi
Dept of Computer Engg and IT
College of Engineering, Pune (CoEP)
Pune, India
gandhias18.comp@coep.ac.in

Vaishnavi Konde
Dept of Computer Engg and IT
College of Engineering, Pune (COEP)
Pune, India
kondevp18.comp@coep.ac.in

Rahul Adhao
Dept of Computer Engg and IT
College of Engineering, Pune (COEP)
Pune, India
rba.comp@coep.ac.in

Vinod Pachghare
Dept of Computer Engg and IT
College of Engineering, Pune (COEP)
Pune, India
vkp.comp@coep.ac.in

2022 International Conference for Advancement in Technology (ICONAT) | 978-1-6654-2577-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICONAT53423.2022.9726105

***Abstract*—The Intrusion Detection System (IDS) - is one that monitors network traffic to issue alerts about any suspicious activity on the network. Conventionally, there are two types of IDSs - Signature-Based, which efficiently detect already known attacks, and Anomaly-Based, where models are trained to detect unknown attacks. The latter type of IDS plays a crucial role in detecting zero-day attacks- a type of attack where the vulnerability of the software is exploited before a developer can take action on it. However, it comes with a few problems, like its high false-positive rates that cause the network to slow down and require constant human intervention and its inability to detect attacks in real-time. This paper analyzes state-of-the-art models that deal with this problem, analyzing their benefits and shortcomings. Further, we propose a framework for addressing these zero-day attacks and reducing their false positive rate of detection using a combination of feature selection methods and fine-tuning of the dataset specifically for false-positive detection. These methods will be tried with various optimizers and models several times, and their results will be compared. We attach results from preliminary testing on the novel idea of a subset of the dataset, with promising results to be applied to find the model that works better than most existing.**

***Index Terms*—Zero-Day Attacks, Feature Selection, Deep Learning, Hybrid Model IDS**

I. Introduction

There has been an exponential increase in cyber-attacks in the past few years. Of these, major harm is done by zero-day attacks because it takes months, sometimes even years, to detect these attacks and fix the vulnerabilities causing them. Zero-day attacks are called so, referring to the idea that a developer has just learned about the attack, hence, has ”zero days” to fix it. These come in many forms- botnet attacks, SSH brute force attacks, DOS, DDOS, infiltration, SQL Injection, and many others - all injected to slow the network, get sensitive information out of the system and commit financial fraud or data fraud. Sometimes, vulnerabilities exist in older versions of the systems, and by the time it is even discovered, hackers use this time to write malicious code and compromise the systems. For an everyday user, attacks can come in several forms like phishing emails, advertisements, or exploitation of a non-updated device. These attacks can exploit vulnerabilities in web browsers, operating systems, office applications, and open-source systems - showing how no one is truly safe from these, people and companies alike. The biggest of these, like Stuxnet, can compromise Iran’s Nuclear Program and Microsoft program vulnerabilities. Table I represents some recent zero-day attacks and the vulnerabilities exploited during these attacks. Much research has been done to curb the effect of these attacks and to avoid them in the first place. Signature-based models have been known to be fast and have a high detection rate- but they can only detect patterns that are already known. Supervised learning methods like KNNs, SVMs, RNNs, Decisions Trees, and unsupervised deep learning methods like ZDAR systems and autoencoder models have been used for anomaly detection. Even though they have shown a good success rate for known attack types, they suffer from a high false-positive rate for unknown attack types. Some hybrid models have combined the two approaches to get the best of signature and anomaly models, but still suffer from a high false-positive rate or a slow overall system. In the domain of cybersecurity, false positives and false negatives are both harmful for their own reasons. False negatives - when an attack is labeled a non-attack, endangers the system and exposes it to malware. The deep learning models introduced have shown a low false-negative rate in the past. However, false-positive rates - when a non-attack is labeled as an attack, have continued to be a problem for this industry. This is harmful because constant human intervention is required to check if the labeled attack is actually an attack. This costs companies in terms of hours and money, and drains the system from reaching optimum speed. All these are important factors when considering the increased internet and software usage due to covid-19. Users and companies alike are now looking for increased speed, which becomes difficult with bulky IDS models with such problems. In this paper, we propose several ways to decrease the false positive rate, including feature selection and using our own fine-tuned dataset, which is a subset of the dataset with false-positive and true-positive classified packets from the confusion matrix. We will be using various models, feature selection methods, adjusting the weights, and trying to fine-tune several times to see at what point the

978-1-6654-2577-3/22/$31.00 ©2022 IEEE

false positive rate of attack detection is reduced while still maintaining an acceptable speed and an acceptable false-negative detection rate.

II. Literature review

The related works have been categorized into the following domains:

A. Signature-based detection intrusion detection systems

Although quite an old model, signature-based detection is still widely used because of its accuracy and speed in detecting known attacks. The paper [1] explores the relevance of a signature-based detection system in today's time for detecting known and unknown attacks. The paper [2] proposes signature extraction for High Volume Attacks (HVAs) and Low Volume Attacks (LVAs). It also makes use of a flash detector to reduce false alarm rates. However, the signature-based detection systems fail to detect new zero-day attacks, which demand the use of Machine Learning and Deep Learning modules for detection.

B. Machine Learning-based intrusion detection systems

Extensive research has been done on the use of various supervised machine learning models for zero-day attack detection. The paper [3] implements six machine learning classification models, namely Random forest classifier, Gaussian naive Bayes classifier, Decision tree classifier, Multi-layer Perceptron (MLP) classifier, K-nearest neighbors (KNN) classifier, and Quadratic discriminant analysis classifier for training the model and after experimentation, concludes that decision tree classifier provides the most accurate results for maximum types of attacks. The paper [4] makes use of SVM for an anomaly-based intrusion detection system because of its robustness. Random Forest classifier is another robust classifier usually used to overcome overfitting and has been implemented in [5]. The reference [6] uses a novel approach by creating a three-phase model to reduce false positives and improve the overall accuracy. An SVM classifier is used in the first phase, a decision tree for the second phase, and the third phase uses naive Bayes. Combining these three classifiers works wonders to improve the accuracy of the IDS.

C. Deep learning-based intrusion detection systems

The increasing complexity of the attacks has prompted people to venture out into deep learning for intrusion detection. Convolutional Neural Networks (CNN), recurrent neural networks (RNN), and auto-encoders are the most common models used. The paper [7] compares four algorithms - DOC, DOC++, OpenMax, and autoSVM - as deep novelty-based classifiers and finds that DOC++ gave the most promising results. This approach helped them not only identify but also classify the unknown attacks. Papers [8] and [9] use a CNN model for classification. The first one used three different internal depths of CNN and concluded that the depth did not significantly affect the performance. The second paper implemented a CNN model to detect DoS attacks and compared its performance with an RNN model to gauge its efficiency. The paper [10] combines the capabilities of recurrent neural networks (RNN), convolutional neural networks (CNN), and simple autoencoder ensemble (SAE) used in parallel to get high accuracy rates.

D. Hybrid models for intrusion detection

All the above models have their own set of advantages, and hence, quite a few attempts have been made to combine different techniques to get the best possible results. The paper [11] proposes a system that combines anomaly-based detection, behavior-based detection, and signature-based detection techniques. It uses 1-class SVM for anomaly detection and Static Analysis Engine (SAE) and Dynamic Analysis Engine (DAE) for malware analysis. The proposed framework in [12] is a combination of signature-based detection and behavior-based detection techniques. A hybrid IDS, which first detects known attacks using a signature-based detection system called Suricata and then uses the isolation forest algorithm (IFA) to detect unknown attacks by discovering network anomalies, is proposed in [13]. The paper [14] proposes a hybrid intrusion detection method that combines anomaly detection using Fuzzy C-means (FCM) with misuse detection using Classification and Regression Trees (CART) and Isolation Forest. This combination showed a significant improvement in the accuracy of the results. The paper [15] proposes a hybrid system consisting of signature-based NIDPS for initial detection, honeypot for extracting the signature and updating the rules, and a temporary queue for storing packets. The papers [16] and [17] both use C5.0 Decision Tree and 1-class SVM together to create a hybrid intrusion detection system. The first paper uses this system for detecting Internet of Things attacks, and the dataset used in both papers is different. All these papers strongly suggest that a hybrid intrusion detection system almost always gives better results than any technique used individually.

E. Feature selection in Intrusion detection systems

As most datasets of attacks contain many features, feature selection becomes an integral part of intrusion detection systems, as it helps in better classification during training datasets by selecting the relevant subset of features. Quite a few papers have used various feature selection and feature reduction techniques in their proposed models. [4] makes use of Principal Component Analysis (PCA) for feature dimension reduction. A combination of three feature selection techniques, namely ExtraTreesClassifier, Percentile, and KBest, is implemented in [18]. The accuracy obtained for a model using feature selection is noticeably higher than not using it. The paper [19] performs feature ranking by two algorithms, information gain and correlation. This ranking is then used to select the appropriate subsets of the features. The paper [10] implements the automatic selection of the optimal parameters and number of features using k-fold(k=10) cross-validation and then uses three popular feature selection metrics: Mutual information, Chi-squared, and ANOVA F-value in parallel. The reference [20] deploys a minimum redundancy maximum relevance (MRMR) feature selection algorithm on the Apache Spark platform to determine the most discriminating features of the dataset. Thus, feature selection plays a crucial role in the training phase of an intrusion detection system.

TABLE I
RECENT ZERO-DAY ATTACKS

Attack	Technique used	Vulnerabilities Exploited	Preventive Measures	Year
Microsoft Exchange Attack by HAFNIUM	Authentication is bypassed by the attacker to secure access to the Exchange Server	Server Side Request Forgery(SSRF), insecure deserialization vulnerability, post-authentication arbitrary file write vulnerability	Proper implementation of Web Application Firewall, Detecting attack quickly by using Collective Defense operational model [21]	March 2021
Kaseya VSA server Attack by REvil	Exploitation of flaw in authentication logic, uploading ransomware and ASP payload	Authentication vulnerability, unrestricted upload of a file, Cross Site Request Forgery(CSRF), code injection vulnerability	Carry out regular patching of systems, do not grant access to vulnerable ports [22]	July 2021
SolarWinds Serv-U attack by Dev-0322	Pre-auth remote code execution on Serv-U servers	Secure Shell (SSH) vulnerability, Address Space Layout Randomization (ASLR) protection missing	ASLR protection should be enabled for binaries [23]	July 2021

III. Proposed architecture

Most state-of-the-art methods still suffer from the problem of a high false-positive rate for detecting various types of zero-day attacks. We propose a series of methods that minimize this while still keeping the network running in an efficient time and keeping the false negatives to a minimum. [6] explores the idea of running SVMs and Decision Trees to ensure that their benefits are retained in the model one after another and considering only the packets with a high confidence rate in both models. We propose a new architecture to reduce false positives that combine feature selection methods, a hybrid model, and a fine-tuned model as described in Fig. 1.

A. Feature selection

Feature selection is one of the most important steps of the training algorithm, as it lets one select the features most suited for one's model. There are several feature selection methods like Pearson coefficient correlation, chi-squared score, and ANOVA. However, using a single selected feature might bias the model towards specific features, resulting in a lower accuracy rate. Therefore, we will experiment with an ensemble of feature selection methods to determine the most important features to select. The goal is to create an algorithm like [24] and determine the most relevant features for our training set using trial and error on various methods and observing their effects on the accuracy of our detection system.

B. Machine Learning algorithms

It is very important to select the right algorithm for classification in an IDS. Logistic regression is the easiest to implement and interpret and is very fast at classifying unknown records, which is beneficial to keep the model fast. SVMs allow for a fast classification in high-dimensional spaces which is very helpful when classifying between several types of attacks, but decision trees are thorough, versatile and powerful. Hence, we are going to use them individually and together as described in [6] to see which algorithm has the best results.

C. Fine-tuning on the positives

After testing on a small part of the dataset, we will get a confusion matrix that outputs the false positive rate, true positive rate, false-negative rate, and true negative rate. Since we have to reduce the false positive rate, we will now create a subset of the dataset consisting of the positives (TP and FP) from our initial test. We hypothesize that fine-tuning the dataset to recognize parameters that distinguish between false positives and true positives only instead of the entire dataset will reduce the false positives rate because the model is now specifically trained to classify and distinguish nuances between the two. This model will specifically distinguish between a false positive and a true positive result and will be used as phase 2 of our model, which is used on data points with low confidence of being benign. We do not use this model in phase 1 classification and allow packets with high confidence of being benign to pass through the network to avoid network congestion and allow for a fast network speed. The flow of the proposed architecture can be seen in Fig. 2.

D. Optimizers

A key part of the phase 2 model is the use of the correct optimizer. Adjusting the weights and learning rates to reduce the losses is very critical in a classification problem. Algorithms like Mini Batch Gradient Descent frequently update the model and have less variance but risk shooting even after reaching the global minima. Adam converges fast and has a high variance but comes with a high computational cost. RMSProp has similar issues and advantages as ADAM but converges at a slower rate. Nesterov-accelerated Adaptive Moment Estimation (NADAM) has an improved momentum from adam and is used heavily in current systems. We will experiment all four in our dataset to see which one can optimize the classification between the positives the best and what advantages/disadvantages each one has.

E. Repetition

Trying phase 2 repeatedly for a certain number of times will give us the optimum false-positive rate without overfitting or increasing the false-negative rate. This is a number that we can find out by experimenting and trying this process out multiple times. This is an interesting figure because it tells us potentially how low the false positive rate can get.

IV. Preliminary results

We have tested our approach on a standard dataset to see how the accuracy improves after implementing a two-stage hybrid model. For this, the IDS 2018 Intrusion CSVs (CSE-CIC-IDS2018) Dataset [25] of the day 02-15-2018 was used.

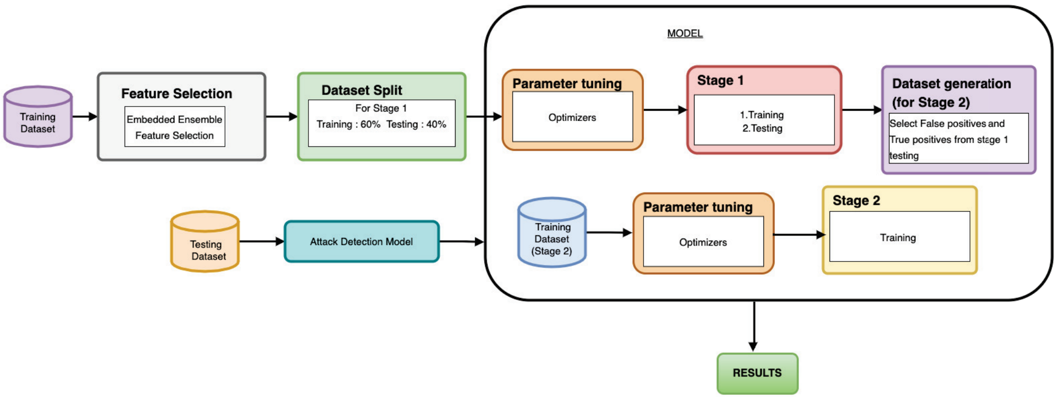

Fig. 1. Steps to be followed for the model in the proposed architecture

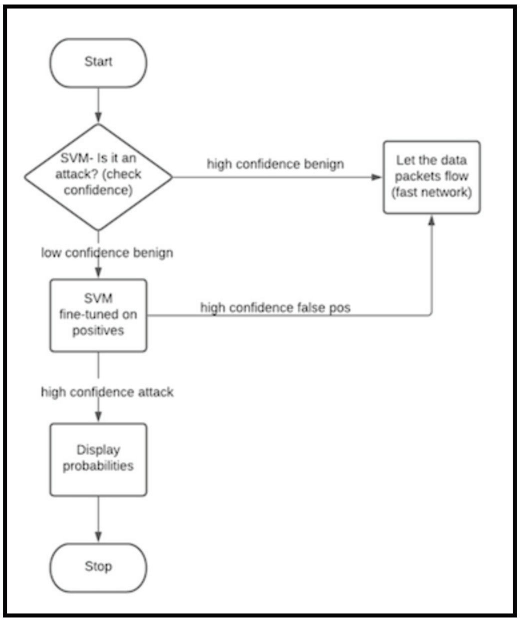
Start
SVM- Is it an attack? (check confidence)
high confidence benign
Let the data packets flow (fast network)
low confidence benign
SVM fine-tuned on positives
high confidence false pos
high confidence attack
Display probabilities
Stop

Fig. 2. Proposed model's data flow

For the basic preprocessing, we took out any data that was not a numeric entry. This eliminated the column 'timestamp'. All the remaining features were taken since we did not want to test the features in this part of the preliminary results, but rather needed an accurate representation of how well the two-staged model performs. In the end, there were 1048575 entries and 79 columns. We took entries from 5000-80000 for stage 1 because there were many malicious entries in the first 10,000 rows, and we needed an equal balance for stages 1 and 2. From these, our data was divided into a train-test split of 0.6-0.4 initially. This gave us 477000 data points for stage 1 and 318000 data points for stage 1 testing (from which we will create our dataset for stage 2 training). We also checked that there were enough malicious entries for stage 1 testing, as we needed to create a dataset from the positives that were left, hence arriving at this split. We implemented Logistic Regression on Stage 1 and got an accuracy of 0.932. The confusion matrix for the same is shown in Table II. We then took the indexes for the positives (true and false) from the test set of 318000 (which gave us around 21,599 values) and applied logistic regression on them, intending to fine-tune the differences between false positives and true positives. Upon testing it on the remaining values from indexes 0:5000 and 80,000 to the end, the accuracy increased to 0.951. The confusion matrix for the same is shown in Table III. Here, we note that the number of false positives has decreased drastically, and while the number of false negatives did increase, the proportion for the two differs drastically. Using feature selection, weighing the attacks more, and using different models, we posit that we will be able to find the appropriate balance, where the false negative values do not increase, but we also get a considerable reduction in false positives.

V. Conclusion

The preliminary results obtained on this dataset show promising results for the two-staged hybrid model. This is particularly important in an IDS which suffers from a huge problem of false positives. Along with the selection of appropriate features through testing and additional testing on different hybrid models, we will be able to find the

TABLE II
CONFUSION MATRIX AFTER STAGE 1

	Predicted: NO	**Predicted: YES**
Actual: YES	TN:99	FP:9
Actual: NO	FN:33	TP:27

TABLE III
CONFUSION MATRIX AFTER STAGE 2

	Predicted: NO	**Predicted: YES**
Actual: YES	TN:99	FP:9
Actual: NO	FN:33	TP:27

best model for the detection of attacks, and increase our accuracy, while decreasing our false-negative rate even more, achieving a model with higher accuracy than most state-of-the-art models.

Acknowledgment

This work was supported by the Information Security Education and Awareness (ISEA Phase II) Project, Department of Electronics and Information Technology, Ministry of Communications and Information Technology, Government of India under Grant No. ISEA-II/PMU/2020 (2020).

References

[1] H. Holm, ”Signature-based intrusion detection for zero-day attacks: (not) a closed chapter?” 2014 47th Hawaii International Conference on System Sciences, 2014, pp. 4895-4904, doi: 10.1109/HICSS.2014.600.

[2] Kumar, V., Sinha, D. “A robust intelligent zero-day cyber-attack detection technique,” Complex Intell. Syst. (2021), doi: 10.1007/s40747-021-00396-9.

[3] Q. Zhou and D. Pezaros, “Evaluation of machine learning classifiers for zero-day intrusion detection – an analysis on CIC-AWS-2018 dataset,” arXiv:1905.03685 [cs.CR].

[4] F. E. Heba, A. Darwish, A. E. Hassanien, and A. Abraham, ”Principle components analysis and Support Vector Machine based Intrusion Detection System,” 2010 10th International Conference on Intelligent Systems Design and Applications, 2010, pp. 363-367, doi: 10.1109/ISDA.2010.5687239.

[5] Nabila Farnaaz, M.A. Jabbar, “Random Forest Modeling for Network Intrusion Detection System,” Procedia Computer Science, Volume 89, 2016, Pages 213-217, ISSN 1877-0509, doi: 10.1016/j.procs.2016.06.047.

[6] K. Goeschel, ”Reducing false positives in intrusion detection systems using data-mining techniques utilizing support vector machines, decision trees, and Naive Bayes for off-line analysis,” SoutheastCon 2016, 2016, pp. 1-6, doi: 10.1109/SECON.2016.7506774.

[7] M. Soltani, B. Ousat, M. J. Siavoshani, A. H. Jahangir “An adaptable deep learning-based Intrusion Detection System to zero-day attacks,” arXiv:2108.09199 [cs.CR]

[8] D. Kwon, K. Natarajan, S. C. Suh, H. Kim, and J. Kim, ”An empirical study on Network Anomaly Detection Using Convolutional Neural Networks,” 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS), 2018, pp. 1595-1598, doi: 10.1109/ICDCS.2018.00178.

[9] Kim, Jiyeon, J. Kim, H. Kim, M. Shim, and Eunjung Choi. 2020. ”CNN-based Network Intrusion Detection against Denial-of-Service attacks,” Electronics 9, no. 6: 916, doi: 10.3390/electronics9060916

[10] C. Lin, A. Li 1 and R. Jiang “Automatic Feature Selection and Ensemble Classifier for Intrusion Detection,” doi: 10.1088/1742-6596/1856/1/012067

[11] R. Kaur and M. Singh, “A hybrid real-time zero-day attack detection and analysis system,” International Journal of Computer Network and Information Security. 7. 19-31. 10.5815/ijcnis.2015.09.03.

[12] D. Cuppah, Ambrish G. and M. Hanumanthappa (2020) “Design and analysis of a hybrid security framework for zero-day attack,” International Journal of Applied Engineering Research, Volume 14, Number 15, 2019, pp. 140-144 ISSN 0973-4562

[13] Z. Chiba, N. Abghour, K. Moussaid, A. Omri and Md. Rida “Newest collaborative and hybrid network intrusion detection framework based on Suricata and isolation forest algorithm,” SCA ’19: Proceedings of the 4th International Conference on Smart City Applications, October 2019 Article No.: 77, pp. 1-11, doi: 10.1145/3368756.3369061

[14] Gun-Yoon Shin1, Dong-Wook Kim1, Sang-Soo Kim and Myung-Mook Han “Unknown Attack Detection: Combining Relabeling and Hybrid Intrusion Detection,” Computers Materials Continua, 2021, Volume 68, Number 3, pp. 3289-3303, doi: 10.32604/cmc.2021.017502

[15] J. H. Jeong and S. G. Choi, ”Hybrid system to minimize damage by zero-day attack based on NIDPS and HoneyPot,” 2020 International Conference on Information and Communication Technology Convergence (ICTC), 2020, pp. 1650-1652, doi: 10.1109/ICTC49870.2020.9289589.

[16] A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, and A. Alazab, 2019, ”A novel ensemble of hybrid Intrusion Detection System for detecting Internet of Things attacks,” Electronics 8, no. 11: 1210, doi: 10.3390/electronics8111210

[17] A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, and A. Alazab, 2020, ”Hybrid Intrusion Detection System based on the stacking ensemble of C5 Decision Tree Classifier and One-Class Support Vector Machine,” Electronics 9, no. 1: 173, doi: 10.3390/electronics9010173

[18] P. HarshaLatha and R. Mohanasundaram, “A New Hybrid Strategy for Malware Detection Classification with Multiple Feature Selection Methods and Ensemble Learning Methods,” International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-9 Issue-2, December 2019, doi: 10.35940/ijeat.B4666.129219

[19] Akashdeep, I. Manzoor and N. Kumar “A feature reduced intrusion detection system using ANN classifier,” Expert Systems with Applications, Volume 88, 2017, pp. 249-257, ISSN 0957-4174, doi:

10.1016/j.eswa.2017.07.005.

[20] S. C. Pallaprolu, R. Sankineni, M. Thevar, G. Karabatis, and J. Wang, ”Zero-day attack identification in streaming data using semantics and Spark,” 2017 IEEE International Congress on Big Data (BigData Congress), 2017, pp. 121-128, doi: 10.1109/BigDataCongress.2017.25.

[21] “HAFNIUM targeting exchange servers with 0-day exploits,” https://www.microsoft.com/security/blog/2021/03/02/hafnium-targeting-exchange-servers/, 2021, [Online; accessed 11-October-2021].

[22] ”How the Kaseya VSA zero-day exploit worked,” https://www.truesec.com/hub/blog/kaseya-vsa-zero-day-exploit, 2021, [Online; accessed 11-October-2021]

[23] “Chinese hackers behind July 2021 SolarWinds zero-day attacks,” https://therecord.media/chinese-hackers-behind-july-2021-solarwinds-zero-day-attacks/, 2021, [Online; accessed 11-October-2021]

[24] R. Venkatarathinam, V. C. Raj and G. George, “EE-OFSA: An Embedded Ensemble Optimal Feature Selection Algorithm for Building Efficient Intrusion Detection System,” International Journal of Pure and Applied Mathematics, Volume 119, Number 15, 2018, pp. 1445-1459

[25] “IDS 2018 Intrusion CSVs (CSE-CIC-IDS2018),” https://www.kaggle.com/solarmainframe/ids-intrusion-csv, 2020, [Online; accessed 29-October-2021]