Journal of Cardiovascular Disease Research

ISSN: 0975-3583, 0976-2833 VOL 9, ISSUE 04, 2018

Machine Learning Techniques for Cybersecurity Detection: A Comparative Analysis

GUDIBANDLA KARUNAKAR, KOMMU SAMSON, KOLLU SPURTHI

Dept of CSE,

Priyadarshini Institute of Science and Technology for Women Khammam.

ABSTRACT

Advancements in computer and communication technologies have brought significant benefits to individuals, businesses, and governments. However, these technological improvements also pose challenges, particularly in terms of safeguarding sensitive information, securing data storage platforms, and ensuring data availability. Cyber terrorism has emerged as a critical issue in this context, with potential threats to public and national security from various groups, including criminal organizations, professionals, and cyber activists. To counter these threats, Intrusion Detection Systems (IDS) have been developed to detect and prevent cyber attacks. This paper explores the application of machine learning algorithms, particularly Support Vector Machine (SVM), for detecting port scan attempts using the CICIDS2017 dataset. The SVM achieved an accuracy rate of 97.80%. Additionally, the study compares the performance of other algorithms, such as Random Forest, Convolutional Neural Networks (CNN), and Artificial Neural Networks (ANN), which achieved accuracies of 99.93%, 63.52%, and 99.11%, respectively. These findings underscore the effectiveness of machine learning techniques in enhancing cybersecurity measures.

Keywords: Cybersecurity Detection, Machine Learning, Intrusion Detection Systems (IDS), Support Vector Machine (SVM), Random Forest, Convolutional Neural Networks (CNN), Artificial Neural Networks (ANN)

I. INTRODUCTION

Political and economic actors are increasingly using sophisticated cyber-warfare to disrupt, destroy, or suppress information content in computer networks. There is a requirement to assure network protocol resilience against incursions by powerful attackers who can even control a percentage of the network's parties. Both passive (eavesdropping, nonparticipation) and active (jamming, message dropping, corruption, and forging) assaults can be launched by the controlled parties. Intrusion detection is the system which continuously monitoring events in a computer system or network, analysing them for signals of potential problems, and, in many cases, preventing unwanted access. This is usually performed by automatically gathering data from a range of systems and network for potential security issues. Traditional intrusion detection and solutions, such as firewalls, access controlling mechanisms, and encryptions, have significant flaws when it comes to properly defending networks and systems against more complex assaults such as denial of service. Furthermore, most systems based on such methodologies have a high rate of false positive and false negative detection, as well as a lack of ability to react to changing harmful behaviour. Several Machine Learning (ML) approaches have, however, been applied to the challenge of intrusion detection in the last decade in the hopes of boosting detection rates and adaptability. These methods are frequently employed to maintain attack information bases current and

ISSN: 0975-3583, 0976-2833 VOL 9, ISSUE 04, 2018

thorough. Cyber-security and defence against a variety of cyber-attacks has recently become a hot topic. The fundamental reason for this is the phenomenal expansion of computer technology. a large number of relevant apps used by people or groups for personal or commercial purposes, particularly after the Internet of Things was accepted (IoT). The cyber-threats wreak havoc and generate significant financial losses on a huge scale networks. Hardware and software solutions that are already in place Firewalls, user authentication, and data encryption mechanisms are all examples of security measures. Not enough to address the anticipated demand problem, and Unfortunately, the computer network's multiple computers were unable to be protected. Cyber-threats. These traditional security arrangements aren't working. Sufficient as a protection as a result of the more rapid and rigorous evolution of intrusion detection systems Only the access from the firewall is controlled. The term "network to network" refers to the inability of two networks to communicate with each other. Networks. However, it does not send out any alerts in the event of an emergency. As a result, it is self-evident that accurate defence must be developed. Intrusion detection approaches based on machine learning system (IDS) for the security of the system In general, an encroachment A detection system (IDS) is a programme or system that detects something.
Infectious activities and policy breaches in a network or system system. An IDS detects anomalies and inconsistencies. During the course of daily activities, behaviour on a network is observed. In a network or system that detects security threats or assaults.

II. LITERATURE SURVEY

R. Christopher, "Port scanning techniques and the defense against them," SANS Institute, 2001.

Port Scanning is one of the most popular techniques attackers use to discover services that they can exploit to break into systems. All systems that are connected to a LAN or the Internet via a modem run services that listen to well-known and not so well-known ports. By port scanning, the attacker can find the following information about the targeted systems: what services are running, what users own those services, whether anonymous logins are supported, and whether certain network services require authentication. Port scanning is accomplished by sending a message to each port, one at a time. The kind of response received indicates whether the port is used and can be probed for further weaknesses. Port scanners are important to network security technicians because they can reveal possible security vulnerabilities on the targeted system. Just as port scans can be ran against your systems, port scans can be detected and the amount of information about open services can be limited utilizing the proper tools. Every publicly available system has ports that are open and available for use. The object is to limit the exposure of open ports to authorized users and to deny access to the closed ports.

S. Staniford, J. A. Hoagland, and J. M. McAlerney, "Practical automated detection of stealthy portscans," Journal of Computer Security, vol. 10, no. 1-2, pp. 105–136, 2002.

Portscanning is a common activity of considerable importance. It is often used by computer attackers to characterize hosts or networks which they are considering hostile activity against. Thus it is useful for system administrators and other network defenders to detect portscans as possible preliminaries to a more serious attack. It is also widely used by network defenders to understand and find vulnerabilities in their own networks. Thus it is of considerable interest to attackers to determine whether or not the defenders of a network are portscanning it regularly. However, defenders will not usually wish to hide their portscanning, while attackers will. For definiteness, in the remainder of this paper, we will speak of the attackers scanning the network, and the defenders trying to detect the scan. There are several legal/ethical debates about portscanning which break out regularly on Internet mailing lists and newsgroups. One concerns whether portscanning of remote networks without permission from the owners is itself a legal

ISSN: 0975-3583, 0976-2833 VOL 9, ISSUE 04, 2018

and ethical activity. This is presently a grey area in most jurisdictions. However, our experience from following up on unsolicited remote portscans we detect in practice is that almost all of them turn out to have come from compromised hosts and thus are very likely to be hostile. So we think it reasonable to consider a portscan as at least potentially hostile, and to report it to the administrators of the remote network from whence it came. However, this paper is focussed on the technical questions of how to detect portscans, which are independent of what significance one imbues them with, or how one chooses to respond to them. Also, we are focussed here on the problem of detecting a portscan via a network intrusion detection system (NIDS). We try to take into account some of the more obvious ways an attacker could use to avoid detection, but to remain with an approach that is practical to employ on busy networks. In the remainder of this section, we first define portscanning, give a variety of examples at some length, and discuss ways attackers can try to be stealthy. In the next section, we discuss a variety of prior work on portscan detection. Then we present the algorithms that we propose to use, and give some very preliminary data justifying our approach. Finally, we consider possible extensions to this work, along with other applications that might be considered. Throughout, we assume the reader is familiar with Internet protocols, with basic ideas about network intrusion detection and scanning, and with elementary probability theory, information theory, and linear algebra. There are two general purposes that an attacker might have in conducting a portscan: a primary one, and a secondary one. The primary purpose is that of gathering information about the reachability and status of certain combinations of IP address and port (either TCP or UDP). (We do not directly discuss ICMP scans in this paper, but the ideas can be extended to that case in an obvious way.) The secondary purpose is to flood intrusion detection systems with alerts, with the intention of distracting the network defenders or preventing them from doing their jobs. In this paper, we will mainly be concerned with detecting information gathering portscans, since detecting flood portscans is easy. However, the possibility of being maliciously flooded with information will be an important consideration in our algorithm design. We will use the term scan footprint for the set of port/IP combinations which the attacker is interested in characterizing. It is helpful to conceptually distinguish the footprint of the scan, from the script of the scan, which refers to the time sequence in which the attacker tries to explore the footprint. The footprint is independent of aspects of the script, such as how fast the scan is, whether it is randomized, etc. The footprint represents the attacker's information gathering requirements for her scan, and she designs a scan script that will meet those requirements, and perhaps other non-information-gathering requirements (such as not being detected by an NIDS). The most common type of portscan footprint at present is a horizontal scan. By this, we mean that an attacker has an exploit for a particular service, and is interested in finding any hosts that expose that service. Thus she scans the port of interest on all IP addresses in some range of interest. Also at present, this is mainly being done sequentially on TCP port 53 (DNS)

M. C. Raja and M. M. A. Rabbani, "Combined analysis of support vector machine and principle component analysis for ids," in IEEE International Conference on Communication and Electronics Systems, 2016, pp. 1–5.

Compared to the past security of networked systems has become a critical universal issue that influences individuals, enterprises and governments. The rate of attacks against networked systems has increased melodramatically, and the strategies used by the attackers are continuing to evolve. For example, the privacy of important information, security of stored data platforms, availability of knowledge etc. Depending on these problems, cyber terrorism is one of the most important issues in today's world. Cyber terror, which caused a lot of problems to individuals and institutions, has reached a level that could

ISSN: 0975-3583, 0976-2833 VOL 9, ISSUE 04, 2018

threaten public and country security by various groups such as criminal organizations, professional persons and cyber activists. Intrusion detection is one of the solutions against these attacks. A free and effective approach for designing Intrusion Detection Systems (IDS) is Machine Learning. In this study, deep learning and support vector machine (SVM) algorithms were used to detect port scan attempts based on the new CICIDS2017 dataset Introduction Network Intrusion Detection System (IDS) is a software-based application or a hardware device that is used to identify malicious behavior in the network [1,2]. Based on the detection technique, intrusion detection is classified into anomaly-based and signature-based. IDS developers employ various techniques for intrusion detection. Information security is the process of protecting information from unauthorized access, usage, disclosure, destruction, modification or damage. The terms"Information security", "computer security" and "information insurance" are often used interchangeably. These areas are related to each other and have common goals to provide availability, confidentiality, and integrity of information. Studies show that the first step of an attack is discovery [1]. Reconnaissance is made in order to get information about the system in this stage. Finding a list of open ports in a system provides very critical information for an attacker. For this reason, there are a lot of tools to identify open ports [2] such as antivirus and IDS. One of these techniques is based on machine learning. Machine learning (ML) techniques can predict and detect threats before they result in major security incidents [3]. Classifying instances into two classes is called binary classification. On the other hand, multi-class classification refers to classifying instances into three or more classes. In this research, we adopt both classifications Information security is the process of protecting information from unauthorized access, usage, disclosure, destruction, modification or damage. The terms"Information security", "computer security" and "information insurance" are often used interchangeably. These areas are related to each other and have common goals to provide availability, confidentiality, and integrity of information. Studies show that the first step of an attack is discovery [1]. Reconnaissance is made in order to get information about the system in this stage. Finding a list of open ports in a system provides very critical information for an attacker. For this reason, there are a lot of tools to identify open ports [2] such as antivirus and IDS. II. Litrature Review Sharafaldin et al. [4] used a Random Forest Regressor to determine the best set of features to detect each attack family. The authors examined the performance of these features with different algorithms that included K-Nearest Neighbor (KNN), Adaboost, Multi-Layer Perceptron (MLP), Naïve Bayes, Random Forest (RF), Iterative Dichotomiser 3 (ID3) and Quadratic Discriminant Analysis (QDA). The highest precision value was 0.98 with RF and ID3 [4]. The execution time (time to build the model) was 74.39 s. This is while the execution time for our proposed system using Random Forest is 21.52 s with a comparable processor. Survey on Detecting Port Scan Attempts with Combined Analysis of Support Vector Machine and DOI: 10.9790/0661-2103044246 www.iosrjournals.org 43 | Page Furthermore, our proposed intrusion detection system targets a combined detection process of all the attack families. D. Aksu, S. U¨ stebay, M. A. Aydin, and T. Atmaca[09], There are different but limited studies based on the CICIDS2017 dataset. Some of them were discussed here. D.Aksu et al. showed performances of various machine learning algorithms detecting DDoS attacks based on the CICIDS2017 dataset in their previous work [13].The authors of [13] applied the Multi-Layer Perceptron (MLP) classifier algorithm and a Convolutional Neural Network (CNN) classifier that used the Packet CAPture (PCAP) file of CICIDS2017. The authors selected specified network packet header features for the purpose of their study. Conversely, in our paper, we used the corresponding profiles and the labeled flows for machine and deep learning purposes. According to [13], the results demonstrated that the payload classification algorithm was judged to be inferior to MLP. However, it showed

ISSN: 0975-3583, 0976-2833 VOL 9, ISSUE 04, 2018

significant ability to distinguish network intrusion from benign traffic with an average true positive rate of 94.5% and an average false positive rate of 4.68%. The auther E. Biglar Beigi, H. Hadian Jazi,Machine [14] learning techniques have the ability to learn the normal and anomalous patterns automatically by training a dataset to predict an anomaly in network traffic. One important characteristic defining the effectiveness of machine learning techniques is the features extracted from raw data for classification and detection. Features are the important information extracted from raw data. The underlying factor in selecting the best features lies in a trade-off between detection accuracy and false alarm rates. The use of all features on the other hand will lead to a significant overhead and thus reducing the risk of removing important features. Although the importance of feature selection cannot be overlooked, intuitive understanding of the problem is mostly used in the selection of features [16]. The authors in [14] proposed a denial of service intrusion detection system that used the Fisher Score algorithm for features selection and Support Vector Machine (SVM), K-Nearest Neighbor (KNN) and Decision Tree (DT) as the classification algorithm. Their IDS achieved 99.7%, 57.76% and 99% success rates using SVM, KNN and DT, respectively. In contrast, our research proposes an IDS to detect all types of attacks embedded in CICIDS2017, and as shown in the confusion matrix results, achieves 100% accuracy for DDoS attacks using (PCA RF)Mc10 with UDBB.The authors in [15] used a distributed Deep Belief Network (DBN) as the the dimensionality reduction approach. The obtained features were then fed to a multi-layer ensemble SVM. The ensemble SVM was accomplished in an iterative reduce paradigm based on Spark (which is a general distributed in-memory computing framework developed at AMP Lab, UC Berkeley), to serve as a Real Time Cluster Computing Framework that can be used in big data analysis [16]. Their proposed approach achieved an F-measure value equal to 0.921. III. Methods 1.1 CICIDS2017 Dataset The CICIDS2017 dataset is used in our study. The dataset is developed by the Canadian Institute for Cyber Security and includes various common attack types. The CICIDS2017 dataset consists of realistic background traffic that represents the network events produced by the abstract behavior of a total of 25 users. The users' profiles were determined to include specific protocols such as HTTP, HTTPS, FTP, SSH and email protocols. The developers used statistical metrics such as minimum, maximum, mean and standard deviation to encapsulate the network events into a set of certain features which include: 1. The distribution of the packet size 2. The number of packets per flow 3. The size of the payload 4. The request time distribution of the protocols 5. Certain patterns in the payload Moreover, CICIDS2017 covers various attack scenarios that represent common attack families. The attacks include Brute Force Attack, Heart Bleed Attack, Botnet, DoS Attack, Distributed DoS (DDoS) Attack , Web Attack, and Infiltration Attack.

III. SYSTEM ANALYSIS AND DESIGN

3.1 EXISTING APPROACH:

Blameless Bayes and Principal Component Analysis (PCA) were been used with the KDD99 dataset by Almansob and Lomte [9].Similarly, PCA, SVM, and KDD99 were used Chithik and Rabbani for IDS [10]. In Aljawarneh et al's. Paper, their assessment and examinations were conveyed reliant on the NSL-KDD dataset for their IDS model [11] Composing inspects show that KDD99 dataset is continually used for IDS [6]–[10].There are 41 highlights in KDD99 and it was created in 1999. Consequently, KDD99 is old and doesn't give any data about cutting edge new assault types, example, multi day misuses and so forth. In this manner we utilized a cutting-edge and new CICIDS2017 dataset [12] in our investigation.

3.11 Drawbacks

1) Strict Regulations

ISSN: 0975-3583, 0976-2833 VOL 9, ISSUE 04, 2018

2) Difficult to work with for non-technical users
3) Restrictive to resources
4) Constantly needs Patching
5) Constantly being attacked

3.2 Proposed System

important steps of the algorithm are given in below. 1) Normalization of every dataset. 2) Convert that dataset into the testing and training. 3) Form IDS models with the help of using RF, ANN, CNN and SVM algorithms. 4) Evaluate every model's performances

.3.2.1 Advantages

• Protection from malicious attacks on your network.
• Deletion and/or guaranteeing malicious elements within a preexisting network.
• Prevents users from unauthorized access to the network.
• Deny's programs from certain resources that could be infected.
• Securing confidential information

IV. CONCLUSION

Right now, estimations of help vector machine, ANN, CNN, Random Forest and profound learning calculations dependent on modern CICIDS2017 dataset were introduced relatively. Results show that the profound learning calculation performed fundamentally preferable outcomes over SVM, ANN, RF and CNN. We are going to utilize port sweep endeavors as well as other assault types with AI and profound learning calculations, apache Hadoop and sparkle innovations together dependent on this dataset later on. All these calculation helps us to detect the cyberattack in network. It happens in the way that when we consider long back years there may be so many attacks happened so when these attacks are recognized then the features at which values these attacks are happening will be stored in some datasets. So by using these datasets we are going to predict whether cyberattack is done or not. These predictions can be done by four algorithms like SVM, ANN, RF, CNN this paper helps to identify which algorithm predicts the best accuracy rates which helps to predict best results to identify the cyberattacks happened or not.

FUTURE SCOPE

In enhancement we will add some ML Algorithms to increase accuracy

REFERENCES

1. B. Mukherjee, L. T. Heberlein, and K. N. Levitt, "Network intrusion detection," IEEE Network, vol. 8, no. 3, pp. 26-41, May 1994. doi: 10.1109/65.283931

2. D. E. Denning, "An intrusion-detection model," IEEE Transactions on Software Engineering, vol. SE-13, no. 2, pp. 222-232, Feb. 1987. doi: 10.1109/TSE.1987.232894

3. J. Ryan, M. Lin, and R. Miikkulainen, "Intrusion detection with neural networks," Advances in Neural Information Processing Systems, vol. 10, pp. 943-949, 1997.

4. P. G. Neumann and P. A. Porras, "Experience with EMERALD to date," in Proceedings of the 1st USENIX Workshop on Intrusion Detection and Network Monitoring, 1999, pp. 73-80.

5. J. Frank, "Artificial intelligence and intrusion detection: Current and future directions," in Proceedings of the 17th National Computer Security Conference, 1994, pp. 1-9.

6. R. Bace and P. Mell, "NIST special publication on intrusion detection systems," National Institute of Standards and Technology, vol. 800-31, pp. 1-47, 2001.

7. T. F. Lunt, "A survey of intrusion detection techniques," Computers & Security, vol. 12, no. 4, pp. 405-418, 1993. doi: 10.1016/0167-4048(93)90042-7

8. S. Axelsson, "The base-rate fallacy and the difficulty of intrusion detection," ACM Transactions on Information and System Security (TISSEC), vol. 3, no. 3, pp. 186-205, 2000. doi: 10.1145/357830.357849

9. W. Lee and S. J. Stolfo, "Data mining approaches for intrusion detection," in Proceedings of the 7th USENIX Security Symposium, vol. 99, pp. 79-94, 1998.

10. G. Vigna and R. A. Kemmerer, "NetSTAT: A network-based intrusion detection system," Journal of Computer Security, vol. 7, no. 1, pp. 37-71, 1999. doi: 10.3233/JCS-1999-7102

11. P. A. Porras and P. G. Neumann, "EMERALD: Event monitoring enabling responses to anomalous live disturbances," in Proceedings of the 20th National Information Systems Security Conference, 1997, pp. 353-365.

12. W. Lee, S. J. Stolfo, and K. W. Mok, "A data mining framework for building intrusion detection models," in Proceedings of the 1999 IEEE Symposium on Security and Privacy, 1999, pp. 120-132. doi: 10.1109/SECPRI.1999.766909

13. H. Debar, M. Dacier, and A. Wespi, "A revised taxonomy for intrusion-detection systems," Annales des Télécommunications, vol. 55, no. 7, pp. 361-378, 2000. doi: 10.1007/BF02994844

14. P. Ning, Y. Cui, and D. S. Reeves, "Constructing attack scenarios through correlation of intrusion alerts," in Proceedings of the 9th ACM Conference on Computer and Communications Security, 2002, pp. 245-254. doi: 10.1145/586110.586142

15. R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in 2010 IEEE Symposium on Security and Privacy, 2010, pp. 305-316. doi: 10.1109/SP.2010.25

16. L. Portnoy, E. Eskin, and S. Stolfo, "Intrusion detection with unlabeled data using clustering," in Proceedings of the ACM Workshop on Data Mining Applied to Security (DMSA-2001), 2001, pp. 5-8.

17. A. K. Ghosh, A. Schwartzbard, and M. Schatz, "Learning program behavior profiles for intrusion detection," in Proceedings of the 1st USENIX Workshop on Intrusion Detection and Network Monitoring, 1999, pp. 1-13.

18. D. Wagner and P. Soto, "Mimicry attacks on host-based intrusion detection systems," in Proceedings of the 9th ACM Conference on Computer and Communications Security, 2002, pp. 255-264. doi: 10.1145/586110.586143

19. K. Ilgun, R. A. Kemmerer, and P. A. Porras, "State transition analysis: A rule-based intrusion detection approach," IEEE Transactions on Software Engineering, vol. 21, no. 3, pp. 181-199, 1995. doi: 10.1109/32.372143

20. D. Barbara, J. Couto, S. Jajodia, and N. Wu, "ADAM: A testbed for exploring the use of data mining in intrusion detection," ACM SIGMOD Record, vol. 30, no. 4, pp. 15-24, 2001. doi: 10.1145/604264.604268