BOTNET ATTACKS DETECTION IN INTERNET OF THINGS USING MACHINE LEARNING

Dr. Nookala Venu[1], Dr.AArun Kumar[2], Mr. A.Sanyasi Rao[3]

[1]Post-Doctoral Research Scholar, Department of Electronics and Communication Engineering, Srinivas University, Mangalore - 574146, India.

[1]Professor, Department of Electronics and Communication Engineering,
Balaji Institute of Technology and Science (Autonomous), Narsampet, Warangal, TS -506331, India,
Email: venunookala@gmail.com

[2]Professor, Department of Computer Science and Engineering,
Balaji Institute of Technology and Science (Autonomous), Narsampet, Warangal, TS -506331, India,
Email: arun.arigala@gmail.com

[3]Associate Professor, Department of Electronics and Communication Engineering,
Balaji Institute of Technology and Science (Autonomous), Narsampet, Warangal, TS -506331, India,
Email: allanki_srao@bitswgl.ac.in

Abstract

The number of Internet-of-Things (IoT) devices has significantly expanded as a result of the growing reliance on the Internet and the associated rise in connectivity demand. According to recent research, the increasing adoption of IoT devices has in turn increased network threats since there are more possible attack surfaces. This demonstrates how IoT networks and devices are becoming more vulnerable and susceptible. As a result, in such situations, proper, effective, and efficient attack detection and mitigation approaches are required. IoT devices' security flaws make it simple for attackers to take advantage of them and incorporate them into a botnet. Once hundreds of thousands of IoT devices have been hijacked and joined a botnet, attackers utilize this botnet to perform sophisticated distributed denial of service (DDoS) assaults that bring down the target websites or services and prevent them from responding to legitimate users. Numerous botnet detection approaches have been put out thus far, but their effectiveness is constrained by the dataset on which they are trained. Due of the variety of attack methods, the features used to train a machine learning(ML) model on a botnet dataset do not perform well on other datasets. Therefore, irrespective of the underlying dataset, we suggest an uniform characteristics set in this work to better detect botnet assaults. When the trained ML models were evaluated over three distinct datasets for botnet attacks, the suggested feature set demonstrated superior results for identifying botnet attacks.

Keywords: IoT, Botnet Detection, Optimization, Security, Botnet attack, DoS attacks, DDoS attacks, Machine Learning(ML).

DOI Number: 10.14704/nq.2022.20.4.NQ22298 **NeuroQuantology 2022; 20(4): 743-754**

1. Introduction

The number of IoT devices has significantly expanded as a result of the growing reliance on the Internet and the associated rise in connectivity demand. Recent predictions that the number of linked devices will reach over 28.5 billion by 2022 [1] and that these IoT devices would be used for a variety of applications, including healthcare [2], smart cities [3], and intelligent transportation systems [4], provide support for this. Because there are more possible attack surfaces as a result of the

eISSN 1303-5150

continuing proliferation of IoT devices, there have been more network assaults. This is supported by a recent Forbes article that claims there were more than 2.9 billion events in 2019, a threefold rise from 2018 [5]. This demonstrates how IoT networks and devices are becoming more vulnerable and susceptible. With academics looking into and proposing more and more possible methods, there is a critical need for appropriate, effective, and fast attack detection and mitigation strategies. Since there is so much data produced by and usable for IoT devices and networks, ML has come to be seen as one potential answer. Systems can be dynamic and adaptable to new inputs thanks to ML, as these systems can "learn" without being explicitly instructed [6]. Additionally, different applications and use cases have shown how effective and efficient ML approaches are [7-12]. They therefore have a great chance of being used for intrusion detection in IoT contexts.

We currently live in a time when real-world objects are becoming smarter, more intelligent, linked to the internet, and capable of communicating with one another without the need for human interaction in order to make life easier and more opulent for people [13]. The idea of the internet of things was introduced when physical devices were connected to the internet. IoT is a communication paradigm in which a variety of everyday things are linked together through the internet. Smart gadgets have become a part of our daily lives as a result of IoT development. Smart home, smart workplace, smart grid, smart healthcare, smart agriculture, smart transportation, and smart city are just a few of the cutting-edge uses of smart technology that have been transformed by IoT[14-18]. Our lives would be greatly impacted by these applications, making smart gadgets more necessary for human existence. Security remains the primary IoT worry, notwithstanding the beneficial revolution. The IoT suppliers want to get their product on the market as soon as possible in order to make more money, hence they are putting more emphasis on device functionality than security. As a result, IoT device security is disregarded. Due to the extensive use of smart devices in our everyday life, a security compromise involving IoT devices would severely impact our way of life. Critical infrastructures including power plants, water plants, nuclear plants, transportation systems, healthcare systems, etc. are also using IoT devices and apps to make them more efficient and remotely accessible. Since IoT devices come with certain major cyber security(CS) issues, the proliferation of their use in critical infrastructures is also raising the possibility of CS attacks[19]. Therefore, if the IoT devices and apps are not properly protected, any hack at these vital infrastructures might have disastrous effects. IoT devices carry over some severe CS issues including shoddy security settings, hardcoded passwords, etc. This is a result of IoT providers' mediocre attention to robust security measures. IoT device security flaws have made it simple for hackers to hijack IoT devices and exploit them for nefarious purposes like botnet attacks. Botnets are networks of linked computers that have malware on them and are remotely managed by command and control servers [20]. The attackers employ the botnet for evil deeds like spam email distribution, click fraud, distributed denial of service (DDoS) assaults to bring down a web service, etc. Botnets have been around for a while, but as unsecured IoT devices proliferate, they have gotten bigger, more complicated, and more hazardous. Botnet assaults are currently a significant danger to the entire internet. The proliferation of unsecure IoT devices has made it simple for attackers to take advantage of them and turn them into soldiers in a botnet

eISSN 1303-5150

army to carry out extensive nefarious operations. Since the botnet attacks have recently resulted in more big and destructive DDoS attacks on renowned web service providers like GitHub [21], Krebs on Security, etc., they are catastrophic not just for users of IoT devices but also for the rest of the world. The first ever significant and potent DDoS assault to occur using an IoT botnet of far more than 30,000 devices was called Mirai [22]. In Mirai, hackers gained access to thousands of IoT devices, such as security cameras, baby toys, wireless printers, etc., by taking advantage of unprotected ports, default or hard-coded passwords [23]. Similar to this, the recent Echobot attack infected millions of IoT devices by taking use of more than 20 different IoT vulnerabilities. Despite the fact that the current firewall and IDS technologies are relatively mature, they are insufficient for IoT Systems because of their diverse traffic patterns, communication protocols, etc.[24]. In order to effectively defend IoT devices against botnet assaults, intelligent IoT-specific security solutions must be designed and integrated with the current security architecture.

2. Literature Survey

Nearly every aspect of modern life is included in the IoT, from body sensors to cloud computing and a lot more. As a result, IoT facilitates integration continually. It links networks, devices, and people all the time, creating a complex web of dispersed systems. The potential of IoT to communicate between machines and humans elevates the quality of human existence to a new level. Smart grids, smart cities, smart homes, and the Industrial IoT are a few examples[14-18]. Numerous methods have so far been put out for identifying botnet assaults. These methods may be roughly divided into two categories, namely the flow-based method and the graph-based method. A flow-based approach uses the network's flow-based properties to extract botnets. In contrast, the graph-based approach uses the graphical communication topology between nodes to identify botnets.

The investigation of ML methods and algorithms as a practical remedy for network security has increased as a result of the recent spike in computer power. For intelligent transportation networks, Li et al. suggested similar models [25]. In order to identify intrusions in autonomous cars, the authors explicitly created tree-based classification models. In contrast, Injadat et al. [26] suggested an improved Bayesian optimization-based network intrusion detection framework. Their tests revealed that the suggested model had a reduced percentage of false alarms and a greater detection accuracy. Similar to this, Injadt et al. also introduced a multi-stage optimized ML-based intrusion detection framework that decreased computing complexity while also increasing detection accuracy [27]. The use of ML classification algorithms for intrusion detection was also suggested by Salo et al. [28]. To identify hidden assault patterns, the authors suggested combining ensemble feature selection with clustering-enabled classification models.

A strategy to stop Internet of Things (IoT) devices from joining the Mirai botnet was put out by Frank et al. [29]. The suggested method entails installing two scripts-hardening script and detection script—on the end device in order to shield it from the Mirai botnet. The detection script is suggested to determine whether the underlying IoT device is a member of a Botnet or not, while the hardening script is for protecting IoT devices from bot assaults. The effectiveness of combining ML techniques with a neural network for botnet detection was examined by Ryu et al. [30]. They combined neural networks with decision tree and Naive

eISSN 1303-5150

Bayes classifiers in an ensemble, and they came to the conclusion that the suggested technique can identify botnet assaults in network traffic more effectively than individual classifiers. To identify bot assaults, Keisuke et al. [31] created a network anomaly detection system employing a Gaussian model. They used a Gaussian mixture model to classify the network traffic as normal or attacking by using principle component analysis (PCA) to the dataset attributes. Using graphics, Sofiane et al[32] .'s suggested an approach to identify botnet assaults. In order to identify the behavior of the botnet, they first constructed a graph of flow sequences based on the source and destination IP addresses of hosts. They then used the unsupervised model to identify outliers in the data. Similar to this, Wang et al[33] .'s two-phase approach for botnet identification was proposed. They employed a flow-based strategy in the initial stage to find a botnet assault. In the second phase, they employed a graph-based approach to track the communication nodes connected to the related botnet nodes. Similar to this, Abbas Abou et al. [34] created a flexible graph to identify botnet assaults. By depicting hosts as nodes and communications between them as vertices, the authors create a communication graph. The majority of the papers mentioned above suggested various ML methods to identify botnet assaults. Some research, including [35-36], and others, presented hybrid feature selection approaches to more effectively identify IoT botnet assaults. The performance of these methods, however, is restricted to the dataset on which they are trained. Due of the variety of attack methods, the characteristics used to train a ML model on a botnet dataset do not perform well on other datasets. So, in this study, we suggested a set of universal criteria that are effective in identifying botnets across all datasets.

3. Methodology

This research suggests a universal characteristics set that is extrapolated based on the Logistic Regression method and frequency counting technique in order to better detect botnet assaults regardless of a dataset. The process as a whole includes six key phases, as seen in Fig. 1, from data collecting through botnet attack detection. Data preprocessing, feature extraction, feature selection, model training, and botnet attack detection are some of these phases.

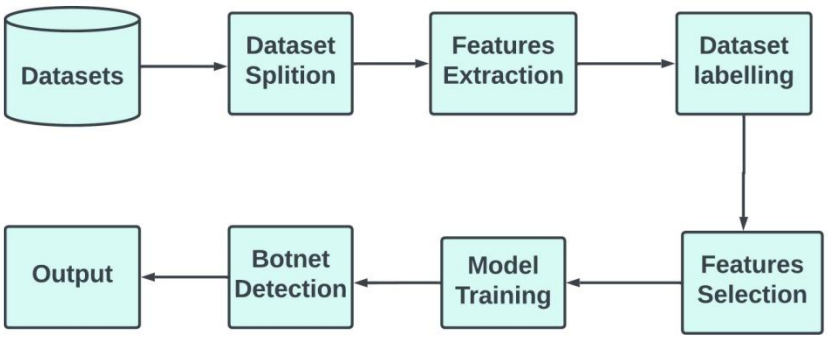


Figure 1. Botnet attack detection: Methodology

3.1 Data Acquisition

The first stage in training a ML model is data collection. Two ways are often used to collect the data. To obtain the data needed to train a ML model, one option is to build up a real-time environment in a lab or to obtain the data from an already-existing real-time environment. Since some firms choose not to disclose their

eISSN 1303-5150

data owing to privacy concerns, not everyone can obtain data from a real-time context. Similar to this, creating a real-time environment in a lab is a difficult task that requires time, resources, and money. Most researchers use the opposite approach, i.e., employ certain publically available datasets for data collecting, to allay these worries. In our situation, we also used the second method and collected data from three freely accessible datasets, namely CICIDS2017, CTU-13, and IoT-23. A recent dataset, CICIDS2017 [37], provides real-time packet captures of several assaults, including the DDoS attack. While the IoT-23 [39] dataset comprises IoT botnet packet captures, the CTU-13 [38] dataset contains classic botnet packet captures. Together with labels provided in text files, all of these datasets are available in packet capture (.pcap) file format. We only took into account the.pcap files from the aforementioned datasets that contain botnet attack traffic and regular network traffic. Due to the fact that the CTU-13 dataset only comprises botnet assaults and typical network traffic, we took into account all of the .pcap files in it. For the CICIDS2017 dataset, we solely took into account the .pcap files from DDoS attack and regular traffic. Finally, we solely took into account the well-known Mirai botnet assault and regular network traffic as recorded in.pcap files from the IoT-23 dataset.

3.2 Data Preprocessing & Features Extraction

This component's objective is to structure the data in a way that will enhance the effectiveness of the created ML classification model. As a result, the min-max approach is used to first normalize the features. To ensure that no one feature predominates throughout the model training phase, the feature normalization step serves to homogenize the dynamic range of the many features. We needed features to train a ML algorithm, however the datasets we got were in .pcap format. In order to go on to the next steps, we must first extract the features from the dataset's .pcap files. The features were extracted from .pcap files using an open-source application called CICFlowmeter [40] for this purpose. The CIC Flow meter utility can extract over 80 features from a given .pcap file, however it has a restriction of being unable to handle files larger than 100MB. Additionally, the obtained datasets include .pcap files that are larger than 100MB.

Dataset Splition: We must perform some data pre-processing before to extracting the features since the CICFlowmeter cannot process .pcap files larger than 100MB. In order to do this, we employed the tcpdump tool [41], which divides larger .pcap files into smaller ones of a certain size. Therefore, we downsized the .pcap files of the datasets with sizes more than 100MB into smaller .pcap files with sizes less than or equal to 100MB using the tcp dump programme.

Features Extraction: We divided the bigger .pcap files into smaller ones, which we then gave to CICFlowmeter to extract the characteristics from. From a given .pcap file, the CICFlowmeter retrieves more than 80 characteristics, which may be divided into two categories: static features and dynamic features. Flow ID, Source IP, Source port, Destination IP, Destination port, and protocol name are among the static properties. While there are roughly 80 flow features in the dynamic features, such as flow duration, packet and byte counts, etc. [42] has a description of these characteristics. Each flow in a given .pcap file has all of these properties taken from it and recorded within a.csv file. These .csv files that were acquired are, however, unlabeled.

Dataset Labeling: The labels for the datasets were provided as text files. We employed SQL server to label these .csv files. In order to

eISSN 1303-5150

distinguish between legitimate and malicious flow as stated by dataset providers, we imported both the .csv files and text files and compared the five tuples, i.e., source IP, source port, destination IP, destination port, and protocol. The .csv files were labeled with the label present throughout the comparable tuple in the text file based on the matching of five-tuple for each flow. All of the datasets were tagged in this manner.

3.3 Features Selection

A ML model's performance is significantly influenced by the characteristics that are chosen [43]. On a dataset, researchers often employ a feature selection strategy before training ML models on the chosen features. Evidently, in this approach, the ML models perform well when evaluated against one dataset but poorly when tested against another [44]. Therefore, regardless of the dataset, we provide a universal feature set that may effectively enable machine algorithms for better botnet attack identification. The process for extrapolating a feature set that may be used globally to detect botnet attacks is shown in Fig. 2. Because it is straightforward, quick, and less complicated than other strategies, we utilized the logistic regression (LR) algorithm for feature selection [45]. After pre-processing Dataset1, we used the LR technique to extract the dataset's top 10 features. On extract the 10 most important characteristics from each dataset, we similarly used the LR technique to Dataset2 and Dataset3. We conducted a frequency analysis after obtaining the relevant features lists from each dataset to determine which characteristics are more commonly used for identifying Botnet attacks across all three datasets. Based on this study, we identified the six traits that are chosen the most frequently across all three datasets. Therefore, we categorized and gave this set of qualities the label "universal features." Finally, in order to train the ML models for identifying botnet assaults, we chose these common properties from each dataset.

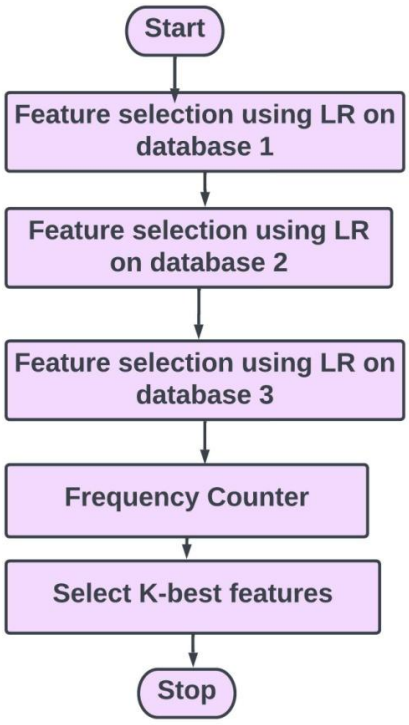


Figure 2. Flowchart

eISSN 1303-5150

3.4Model Training

The ML models must be trained in order to detect botnet assaults after the feature selection. We independently chose the six aforementioned attributes for this purpose from each of the three datasets, namely CICIDS2017, CTU-13, and IoT-23. Then, we divided each dataset into a training and testing set with an 80:20 ratio, meaning that 80% of the data was chosen at random for training and 20% of the data was chosen at random for testing. This data was then used to train the four most popular ML algorithms to recognize botnet and DDoS attacks. These ML methods include Logistic Regression (LR), K-Nearest Neighbors (KNN), Random Forest(RF), and Naive Bayes (NB).

3.5 Botnet Attack Detection

The final stage is to assess the trained model's performance on unobserved data after it has been taught to detect botnet assaults. As previously indicated, we chose 20% of the data to measure how well the trained model performed. We evaluated the performance of each trained model across four parameters after testing it using 20% unseen data in this step.

4. Results

As previously indicated, we trained four widely used ML models, NB, KNN, RF, and LR, for detecting various forms of botnet attacks across three datasets after defining the universal characteristics set for botnet assaults detection. We generated four performance matrices to assess the trained models' performance. Accuracy, precision, recall, and F1-measure are some of these measurements.

Accuracy is the capacity of the system to distinguish between legitimate traffic as a "normal flow" and botnet attacks as a "attack". It provides information on the percentage of accurate predictions across all samples.

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \text{x}100$$

The degree of precision indicates how many of the expected botnet assaults were successful. It is a ratio between the assaults that were accurately predicted (TP) and the actual outcomes (TP+FP).

$$\text{Precision} = \frac{TP}{TP + FP} X100$$

The capacity of the system to accurately identify the botnet assault after a security breach is known as recall. It also goes by the name of sensitivity.

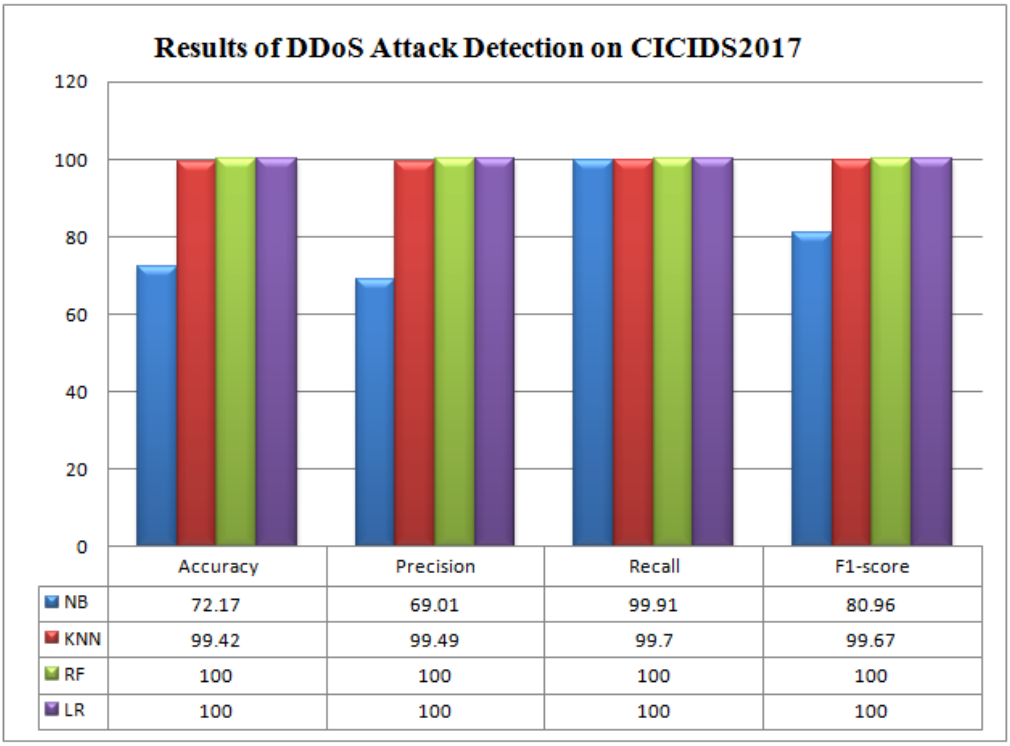
$$\text{Recall} = \frac{TP}{TP + FN} X100$$

The weighted harmonic mean of recall and accuracy is known as the F1-Score. The percentage of accurate predictions in the test set is disclosed.

$$\text{F1-Score} = \frac{2 \text{ x Recall x Precision}}{\text{Recall + Precision}}$$

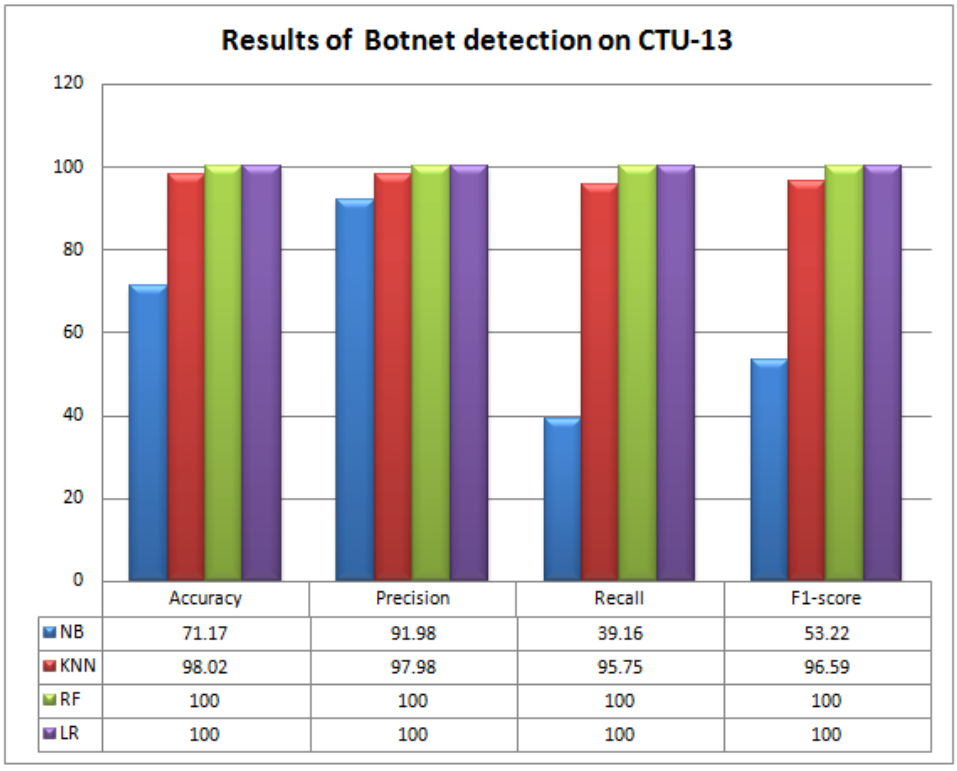
Here TP is the number of occurrences that are true positive, TN is the number of instances that are true negative, FP is the number of instances that are false positive, and FN is the number of instances that are false negative.

eISSN 1303-5150



	Accuracy	Precision	Recall	F1-score
NB	72.17	69.01	99.91	80.96
KNN	99.42	99.49	99.7	99.67
RF	100	100	100	100
LR	100	100	100	100

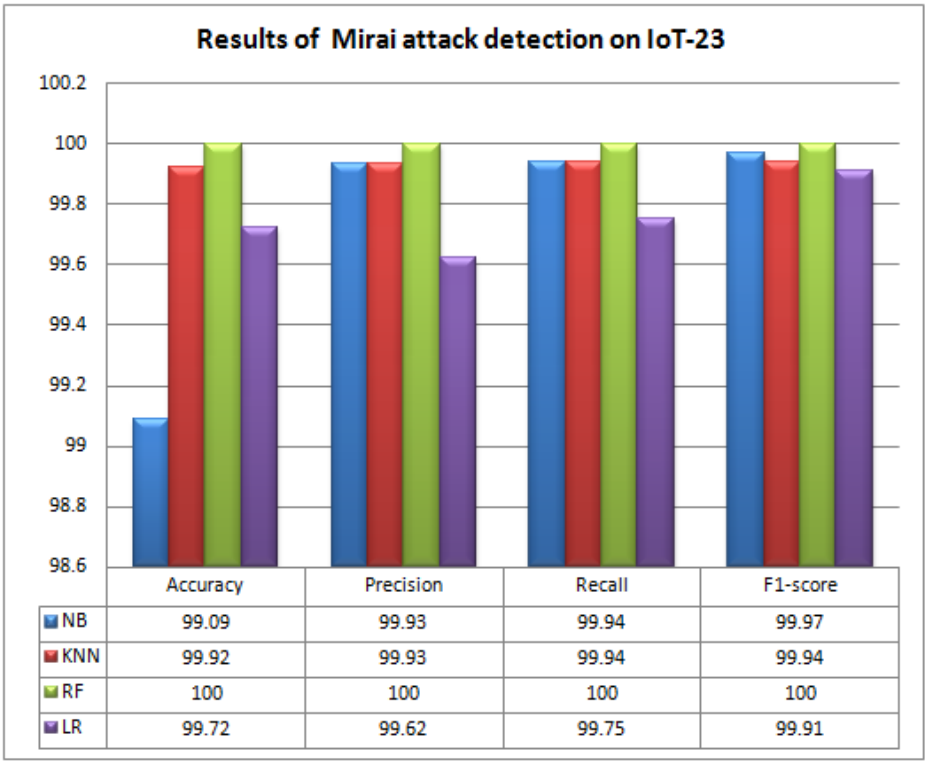
Figure 3. Results of the CICIDS2017 DDoS attack detection



	Accuracy	Precision	Recall	F1-score
NB	71.17	91.98	39.16	53.22
KNN	98.02	97.98	95.75	96.59
RF	100	100	100	100
LR	100	100	100	100

Figure 4. Results of the CTU-13Botnet attack detection

eISSN 1303-5150



	Accuracy	Precision	Recall	F1-score
NB	99.09	99.93	99.94	99.97
KNN	99.92	99.93	99.94	99.94
RF	100	100	100	100
LR	99.72	99.62	99.75	99.91

Figure 5. Results of the IoT-23 Mirai attack detection

5. Conclusion

Due to an increased reliance on the Internet and an increase in connectivity demand, there has been a noticeable development in the deployment of IoT devices in recent years. Recent predictions that the number of linked devices would reach about 40 billion by 2025 are in favour of this. Because there are now more possible attack surfaces, this has increased the number of network attacks. To guarantee that these devices are effectively protected, appropriate, effective, and efficient attack detection and mitigation mechanisms are required. Botnet attacks have grown to be a significant security risk to the internet as the proliferation of vulnerable IoT devices increases. Numerous ML based techniques have thus far been put out for detecting various botnet assaults. The feature set that is utilized to train the ML models has a significant impact on how well these ML-based solutions function. Due to the variety of botnet attacks, characteristics chosen from one specific dataset typically do not assist ML models for effectively identifying the botnet assaults on other datasets. We therefore suggested universal characteristics set in this study to better enable the ML models for identifying various botnet assaults. Four widely used ML algorithms are trained on the suggested characteristics set to identify botnet assaults across three distinct datasets. When trained and tested using the suggested universal feature set spanning three distinct datasets, the testing findings showed that the ML algorithms successfully recognized the botnet assaults.

References

[1] Cisco, “Cisco Predicts More IP Traffic in the Next Five Years Than in the History of the Internet,” Nov. 2018.

[2] Z. Alansari, S. Soomro, M. R. Belgaum, and S. Shamshirband, “The rise of internet of things (iot) in big healthcare data: review and open research issues,” in Progress in Advanced

eISSN 1303-5150

Computing and Intelligent Engineering. Springer, 2018, pp. 675–685.

[3] H. Arasteh, V. Hosseinnezhad, V. Loia, A. Tommasetti, O. Troisi, M. Shafie-khah, and P. Siano, “Iot-based smart cities: A survey,” in 2016 IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC), 2016, pp. 1–6.

[4] I. Al Ridhawi, M. Aloqaily, B. Kantarci, Y. Jararweh, and H. T. Mouftah, “A continuous diversified vehicular cloud service availability framework for smart cities,” Computer Networks, vol. 145, pp. 207–218, 2018.

[5] Z. Doffman, “Cyberattacks on iot devices surge 300% in 2019,‘measured in billions,’report claims,” 2019.

[6] A. Moubayed, M. Injadat, A. B. Nassif, H. Lutfiyya, and A. Shami, “Elearning: Challenges and research opportunities using machine learning data analytics,” IEEE Access, vol. 6, pp. 39 117–39 138, 2018.

[7] A. Moubayed, M. Injadat, A. Shami, and H. Lutfiyya, “Student engagement level in an e-learning environment: Clustering using k-means,” American Journal of Distance Education, vol. 34, no. 2, pp. 137–156, 2020.

[8] “Relationship between student engagement and performance in e-learning environment using association rules,” in 2018 IEEE World Engineering Education Conference (EDUNINE), 2018, pp. 1–6.

[9] M. Injadat, A. Moubayed, A. B. Nassif, and A. Shami, “Systematic ensemble model selection approach for educational data mining,” Knowledge-based Systems, vol. 200, p. 105992, Jul. 2020.

[10] “Multi-split optimized bagging ensemble model selection for multiclass educational data mining,” Applied Intelligence, pp. 1–23, Jul. 2020.

[11] A. Moubayed, M. Injadat, A. Shami, and H. Lutfiyya, “DNS Typo- Squatting Domain Detection: A Data Analytics & Machine Learning Based Approach,” in 2018 IEEE Global Communications Conference (GLOBECOM), Dec. 2018, pp. 1–7.

[12] A. Moubayed, E. Aqeeli, and A. Shami, “Ensemble-based feature selection and classification model for dns typo-squatting detection,” in 2020 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE), Aug. 2020.

[13] H. Xia, L. Li, X. Cheng, X. Cheng, and T. Qiu, “Modeling and analysis botnet propagation in social internet of things,” IEEE Internet of Things Journal, 2020.

[14] K. K. Vaigandla, "Communication Technologies and Challenges on 6G Networks for the Internet: Internet of Things (IoT) Based Analysis," 2022 2nd International Conference on Innovative Practices in Technology and Management (ICIPTM), 2022, pp. 27-31, doi: 10.1109/ICIPTM54933.2022.9753990.

[15] Dr.Nookala Venu, Dr.A.ArunKumar and Karthik Kumar Vaigandla. Review of Internet of Things (IoT) for Future Generation Wireless Communications. International Journal for Modern Trends in Science and Technology 2022, 8(03), pp. 01-08. https://doi.org/10.46501/IJMTST0803001

[16] Karthik Kumar Vaigandla , Radha Krishna Karne , Allanki Sanyasi Rao, " A Study on IoT Technologies, Standards and Protocols", IBM RD's Journal of Management & Research, Volume 10, Issue 2, September 2021, Print ISSN : 2277-7830, Online ISSN: 2348- 5922, DOI: 10.17697/ibmrd/2021/v10i2/166798

[17] KarthikKumar Vaigandla, Nilofar Azmi, RadhaKrishna Karne, "Investigation on Intrusion Detection Systems (IDSs) in IoT," International Journal of Emerging Trends in Engineering Research, Volume 10. No.3, March 2022, https://doi.org/10.30534/ijeter/2022/041032022

[18] Dr.Nookala Venu, Dr.A.ArunKumar, Karthik Kumar Vaigandla, "Investigation on Internet of

Things(IoT) : Technologies, Challenges and Applications in Healthcare," International Journal of Research, Volume XI, Issue II, February/2022, pp.143-153

[19] OWASP Releases Latest Top 10 IoT Vulnerabilities, (accessed April 26, 2020). [Online]. Available: https://www.techwell.com/techwell-insights/2019/01/owasp-releases-latest-top-10-iot-vulnerabilities

[20] T. Alladi, V. Chamola, B. Sikdar, and K.-K. R. Choo, "Consumer iot: Security vulnerability case studies and solutions," IEEE Consumer Electronics Magazine, vol. 9, no. 2, pp. 17–25, 2020.

[21] GitHub Survived the Biggest DDoS Attack Ever Recorded, (accessed April 26, 2020). [Online]. Available: https://www.wired.com/story/github-ddos-memcached

[22] M. S. Pour, A. Mangino, K. Friday, M. Rathbun, E. Bou-Harb, F. Iqbal, S. Samtani, J. Crichigno, and N. Ghani, "On data-driven curation, learning, and analysis for inferring evolving internet-of-things (iot) botnets in the wild," Computers & Security, vol. 91, p. 101707, 2020.

[23] Z. Ahmed, S. M. Danish, H. K. Qureshi, and M. Lestas, "Protecting iots from mirai botnet attacks using blockchains," in 2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD). IEEE, 2019, pp. 1–6.

[24] B. B. Zarpelao, R. S. Miani, C. T. Kawakani, and S. C. de Alvarenga, "A survey of intrusion detection in internet of things," Journal of Network and Computer Applications, vol. 84, pp. 25–37, 2017.

[25] L. Yang, A. Moubayed, I. Hamieh, and A. Shami, "Tree-based intelligent intrusion detection system in internet of vehicles," in 2019 IEEE Global Communications Conference (GLOBECOM), Dec 2019, pp. 1–6.

[26] M. Injadat, F. Salo, A. B. Nassif, A. Essex, and A. Shami, "Bayesian optimization with machine learning algorithms towards anomaly detection," in 2018 IEEE Global Communications Conference (GLOBECOM), Dec 2018, pp. 1–6.

[27] M. Injadat, A. Moubayed, A. B. Nassif, and A. Shami, "Multi-stage optimized machine learning framework for network intrusion detection," IEEE Transactions on Network and Service Management, pp. 1–1, Aug. 2020.

[28] F. Salo, M. Injadat, A. Moubayed, A. B. Nassif, and A. Essex, "Clustering enabled classification using ensemble feature selection for intrusion detection," in 2019 International Conference on Computing, Networking and Communications (ICNC), 2019, pp. 276–281.

[29] C. Frank, C. Nance, S. Jarocki, and W. E. Pauli, "Protecting iot from mirai botnets; iot device hardening," Journal of Information Systems Applied Research, vol. 11, no. 2, p. 33, 2018.

[30] S. Ryu, B. Yang et al., "A comparative study of machine learning algorithms and their ensembles for botnet detection," Journal of Computer and Communications, vol. 6, no. 05, p. 119, 2018.

[31] K. Kato and V. Klyuev, "Development of a network intrusion detection system using apache hadoop and spark," in 2017 IEEE Conference on Dependable and Secure Computing. IEEE, 2017, pp. 416–423.

[32] S. Lagraa, J. Franc¸ois, A. Lahmadi, M. Miner, C. Hammerschmidt, and R. State, "Botgm: Unsupervised graph mining to detect botnets in traffic flows," in 2017 1st Cyber Security in Networking Conference (CSNet). IEEE, 2017, pp. 1–8.

[33] J. Wang and I. C. Paschalidis, "Botnet detection based on anomaly and community detection," IEEE Transactions on Control of Network Systems, vol. 4, no. 2, 2017.

[34] A. A. Daya, M. A. Salahuddin, N. Limam, and R. Boutaba, "A graphbased machine learning approach for bot detection," in 2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM). IEEE, 2019, pp. 144–152.

[35] I. Letteri, G. Della Penna, and P. Caianiello, "Feature selection strategies for http botnet traffic detection," in 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW). IEEE, 2019, pp. 202– 210.

[36] A. Guerra-Manzanares, H. Bahsi, and S. N˜omm, "Hybrid feature selection models for machine learning based botnet detection in iot networks," in 2019 International Conference on Cyberworlds (CW). IEEE, 2019, pp. 324–327.

[37] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization." in 4th International Conference on Information Systems Security and Privacy (ICISSP), 2018, pp. 108–116.

[38] S. Garcia, M. Grill, J. Stiborek, and A. Zunino, "An empirical comparison of botnet detection methods," computers & security, vol. 45, pp. 100–123, 2014.

[39] Stratosphere Laboratory. A labeled dataset with malicious and benign IoT network traffic. Agustin Parmisano, Sebastian Garcia, Maria Jose Erquiaga, (accessed April 26, 2020). [Online]. Available: https://www.stratosphereips.org/datasets-iot23

[40] CICFLOWMETER, (accessed June 30, 2020). [Online]. Available: https://github.com/ahlashkari/CICFlowMeter

[41] WinDump: tcpdump for Windows using WinPcap, (accessed May 18, 2020). [Online]. Available: https://www.winpcap.org/windump/

[42] NETWORK TRAFFIC FLOW ANALYZER, (accessed January 6, 2020). [Online]. Available: http://www.netflowmeter.ca/netflowmeter.html

[43] J. Barnes, R. Klinger, and S. S. im Walde, "Assessing state-of-the-art sentiment models on state-of-the-art sentiment datasets," in Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, 2017, pp. 2–12.

[44] D. Stamos, S. Martelli, M. Nabi, A. McDonald, V. Murino, and M. Pontil, "Learning with dataset bias in latent subcategory models," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3650–3658.

[45] B. K. Dedeturk and B. Akay, "Spam filtering using a logistic regression model trained by an artificial bee colony algorithm," Applied Soft Computing, p. 106229, 2020.