Botnet attacks classification and detection in IoT using Machine learning Techniques

***Abstract*— IoT services and apps have become increasingly popular. Companies are developing IoT-based products, including autonomous driverless cars, a microgrid, smart mineral extraction, and process automation. Data thieves and cyber attackers have been drawn to the volume and prevalence. IoT security is a serious problem. This study uses machine learning to identify and counteract botnet-based Distributed Denial Of Service attacks in IoT networks. Bot security is addressed by our model. Machine learning algorithms like K-Nearest Neighbor , the Naive Bayes model, and the Multi-layer Perception Artificial Neural Network were used to create a model using the BoT-IoT dataset. Analyze the forensics reliability of the BoT-IoT dataset using statistical and machine learning techniques. IoT-specific networks can now identify botnets thanks to this work. Networks must be protected from these devices' access to IoT data. The effectiveness of machine learning, including deep learning, in identifying IoT botnets has been demonstrated in recent studies. For IoT scalability and computation resource problems, minimizing classification features is essential. The ability to quickly interpret data and create signatures for intrusion detection and network monitoring systems is a requirement for cyber security analysts. In this study, feature selection was employed to find IoT bots that had fewer features. A decision tree-based multi-class classifier can achieve high accuracy and understandable results with fewer features.**

***Key Words* —IoT, IoT-threat, Bot-net, middleware, machine learning, DDoS, Dos, Information Theft.**

INTRODUCTION

Internet-of-Things (IoT) devices are susceptible to various cyberattacks due to the fact that they connect to the internet and enable autonomous device to device communication. To guarantee the safety of the IOT network and devices, it is essential that the right security requirements are identified at the outset of the design and deployment processes. Due to the immaturity of the Internet of Things, there is no solid security infrastructure or mechanism in place to protect sensitive data. Security best practices must be applied to the IoT network to ensure the safety of all connected devices, users, and institutions. Distributed Denial of Service attacks using IoT botnets, in which hackers infect the devices with script, are the biggest security risk. In addition to its widespread application in the realms of manufacturing, energy management, transportation, home-automation, and health-monitoring, the IoT has also found use in the fields of education and research.

Data loss, privacy violations, and even physical harm are all possible outcomes of attacks on IoT devices at the physical-network, and application layers.

The compromised devices have a huge potential to increase botnet damage, especially in denial of service(DoS) attacks, which has far reaching consequences for all other information systems. The detection of botnets is just one of many intrusions that have been addressed with the help of machine learning techniques in recent years. On the one hand, the problem's high dimensionality makes it attractive from a deep-learning perspective. However, the issue may be resolved by employing appropriate feature selection and dimensionality reduction methods. It was demonstrated that highly accurate models could be generated using deep learning methods.

RELATED WORK

A "botnet," or complex network of bots, is a tool used by criminals operating online to carry out destructive operations. One of the biggest problems with the IoT is attacks that rely on botnets. Due to the particular requirements, such as low latency and a distributed-nature, the detection of threats in IoT networks is noticeably different. Numerous studies have been conducted in an effort to create efficient botnet detection systems. The early work on botnet identification that used a range of machine learning methods is covered in this section.

To identify distributed denial of service threats, a machine learning technique based on feature engineering was created. In order to compare the performance of different machine learning techniques, the authors combine feature engineering with k-nearest neighbors (KNN), multinomial Naive Bayes (MNB), and random forest (RF) decision trees. The authors employ the chi2 and information gain scores in supervised machine learning methods to acquire the best feature tuning feasible Machine Learning Algorithms. The findings suggested that reducing the number of dimensions utilized to represent the characteristics could enhance the detection model's performance. After putting KNN algorithms to the test on a range of Machine Learning Algorithims, the author finds that they work well. He next looks at the accuracy ratings of datasets with less characteristics. This study demonstrates that feature reduction is feasible while posing little threat to the system's accuracy. By doing this, system processing overhead may be decreased. Similar research on dimensionality reduction for machine learning has also been done by the author, and it may be compared to the earlier study. Reducing the dimensionality of the key features, in the author's opinion, can help with scalability and computing overhead issues. In order to choose the most crucial features to include in the decision-tree algorithm and employ fewer features overall, the study's author used the approach for feature selection. The study provides enlightening figures that unmistakably show how using fewer criteria may result in higher rates of accuracy.

The data collected from IoT honeypots is used by the authors to build a machine learning model. By giving attackers knowledge about the malware used in Bot-net attacks, honeypots are used to persuade them to investigate attack launching techniques. This data was obtained via botnet software. The upcoming machine learning models are trained using these data. To grant an attacker access via an open port, the authors use a honeypot. The ultimate goal is to document every exchange that occurs between our gadget and the assailant. This intercepted file contains details on fresh malware

families. A comparison must be done in order to assess the effectiveness of numerous Machine Learning Algorithms, including random forest, K-nearest neighbors, and decision tree models. Each packet's size, interpacket times, and protocol features are decided by the authors. This method has the benefit of being able to identify recently identified malware families used in Bot-net attacks. The detection decision is a step in a process that involves human specialists working in security operation centers to evaluate events or monitor systems, rather than being a stand-alone activity. These experts take part in the activity. High accuracy classifiers that are challenging to read won't be able to meet the demands of these operational situations. Another thing to bear in mind is that applying the learnings to the infrastructure that already exists for intrusion detection and system monitoring is always preferable. This avoids the need for significant adjustments and further spending.

In a large amount of research that has not mostly focused on Internet of Things networks, machine learning has been used.

The development of feature selection algorithms based on computationally intensive wrapper approaches has improved the accuracy of detection in these networks. Deep autoencoders are a technique for Internet of Things networks that helps with anomaly identification, k-nearest neighbors (KNN), support vector machines (SVM), decision-trees, random-forests are used in a separate line of research that makes use of an Internet of Things dataset.

In this part, here present a brief overview of the botnet, as well as background on the Internet of Things security issues, Bot-net malware, the botnet life-cycle, various Bot-net detection approaches, the concept of machine learning, and the machine learning algorithms used in this project.

Botnet Scenarios:

A botnet is a group of bots used to attack a target network and controlled by a central command and control node, or botmaster. Infected computers, or bots, can be operated maliciously by a botmaster from afar and often leave no traces of their infection. Botnets can be very tiny, consisting of just a few hundred bots, or very large, with more than fifty thousand hosts. Distributing botnet software is a common tactic used by hackers; they can remain undetected for long periods of time, and they can keep producing malware for years. Botnet control nodes are shown in Fig. 1. Botnets can't function without two-way communication between the botmaster and the bots under his command. It is important to communicate with the bots in order to issue commands for them to carry out malicious acts.

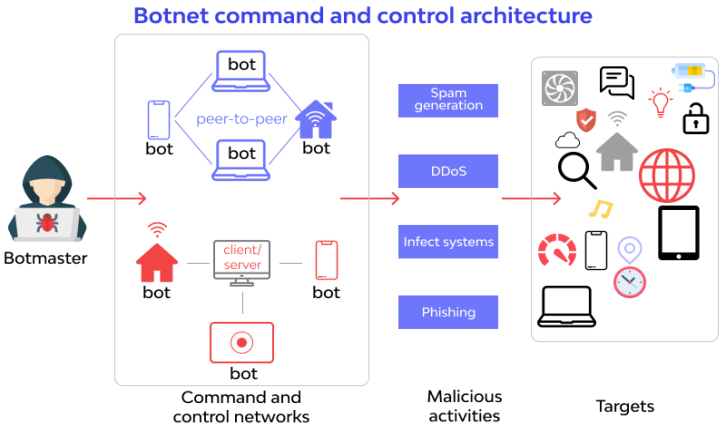

Fig1: Botnet Command and Control architecture

Botmaster maintains a modest profile at all times while providing secret support for the botnet infrastructure. There is always a central command and control server where the botmaster and bots may communicate with one another. The primary goal of most bots is to go unnoticed until they are needed to perform some action. Due to the fact that they do not disrupt the host's normal operations and remain silent until given the instruction by the botmaster to do their activities, hidden bots are harder to detect. The phases of a botnet's life cycle include: infection and spread, secondary-injection, connection, malicious command and control, updates, and maintenance.

Distributed Denial of Service:

DDoS is the most popular cyber-attack in which attackers transmit a massive amount of malicious traffic to the target server at once. Distributed Denial of Service attacks aim to disrupt regular server operation by flooding the target device with large traffic, such as false requests, to oversaturate its capacity and disrupt genuine traffic. DDoS assaults damage the server's Computer processing units, Hardware memory, and network bandwidth, causing genuine computers to be refused service. DDoS attack using Bot-net. Internet of Things devices are implicated in DDoS attacks after being compromised by malicious software. Infected IoT devices are utilized as DDoS bots.

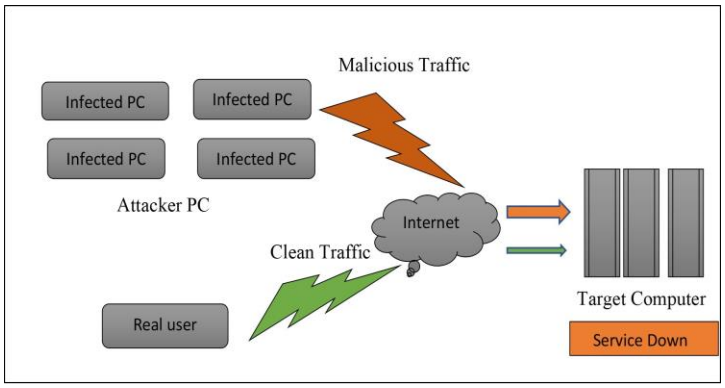

Fig2: DDoS explanation in Block Diagram

Denial Of Service:

Denial of service attacks disrupt a service, making it inaccessible to authorized users. The dataset includes the following DDoS and DoS attacks: Bots are compromised computers that carry out DDoS and DoS attacks. These attacks target a remote server. These attacks disrupt authorized users' services.

These attacks are classed by their method. These gangs use volumetric and protocol-based attacks. A volumetric attack either forces the victim to execute attack-generated requests or brings down the system, making the service unavailable. Volumetric assaults cause network congestion. Protocol based assaults exploit Internet protocol methods. This depletes a system's CPU and memory, preventing it from responding to queries.

Information Theft:

The term "information theft" is used to describe a category of assaults in which the goal is to get unauthorized access to data via weakening a system's defenses. The dataset includes the following classifications of information theft assault types:

Information theft attacks may be broken down into numerous different types based on the attacker's intended outcome. Theft of sensitive data is the first category. Data theft attacks include an off-site computer being infiltrated in order to steal information and send it to the attacker's workstation. The second type of subset is key-loggers. During keylogging activities, an attacker breaks into a remote computer in order to record a user's key-strokes and, perhaps, steal their credentials. To maximize their impact, cyber-criminals frequently combine various forms of information theft with Advanced Persistent Threat techniques.

Information gathering	Service scanning		nmap, hping3	1463364
	OS Fingerprinting		nmap, xprobe2	358275
Denial of Service	DDoS	TCP	hping3	19547603
		UDP	hping3	18965106
		HTTP	golden-eye	19771
	DoS	TCP	hping3	12315997
		UDP	hping3	20659491
		HTTP	golden-eye	29706
Information theft	Keylogging		Metasploit	1469
	Data theft		Metasploit	118
Total				73360900

MACHINE LEARNING

Mathematical models are significantly used in today's security systems; however, they don't always represent the systems' accuracy. Effective wireless network security calls for sophisticated mathematical answers, which take a long time to calculate. Therefore, machine learning algorithms will play a crucial role in IoT security solutions due to their proficiency in modeling systems that cannot be described by mathematical equations. Machine learning is a subfield of computer science that studies how machines may gain knowledge from data and examples. The detection of aberrant traffic that may indicate network intrusion attempts is made possible thanks to a new anomaly detection model based on machine learning. Machine learning algorithms are used to train computers to make judgments without being explicitly programmed by constructing a mathematical model from data samples. There are a total of three types of MLAs, distinguished by the level of guidance they received during their first education. The three forms of education available are called supervised, un-supervised, and semi-supervised learning.

SUPERVISED MACHINE LEARNING ALGORITHMS

KNN

K Nearest Neighbor algorithms works based on Euclidean distance calculation and object is categorized by majority of vote of its K neighbors with the entity of different classes.The value of K is positive and usually small. The accuracy of KNN algorithm depend on the number of neighbors chosen that is the value of K. Usually the value of K is chosen odd number for binary classification to evade the possibility of two classes labels acquiring the same count. If the value of K is chosen 1 then the entity is simply assigned to its single nearestclass. Value of K chosen should be optimal, if the value of Kis small, then it could be under-fitting as well as larger value can cause over-fitting of the model.

Multinomial Naïve Bayes (MNB):

In NLP, the Multinomial Naive Bayes algorithm is a common probabilistic learning strategy (NLP). The computer software that can determine the tag linked with a text such as an email or a newspaper article is based on the Bayes theorem. The possibility of each tag occurring in a given sample is calculated, and the tag with the highest likelihood is chosen as the one to be produced.

The Naive Bayes classifier is a suite of algorithms that have a common philosophy: the characteristics being classified have no inherent relationship to any of the other features. Both characteristics can exist independently of each other, with no causal relationship between them.

Decision Tree:

The supervised learning family of learning algorithms includes the decision tree approach. In contrast to other supervised learning methods, the decision tree methodology may also be used to address classification and regression-related problems.

By learning fundamental decision rules derived from prior data, a Decision Tree is used to create a training model that may be used to predict the class or value of the target variable. Use of the Decision Tree can be used to achieve this (training data). They start at the root of the tree to make our forecasts when using Decision Trees to anticipate a class label for a record. We contrast the root attribute's value with the value of the record's attribute. After following the branch that corresponds to the value in question, we proceed to the next node as suggested by the comparison.

Random Forest:

In classification and regression problems, supervised machine learning methods like as random forest are widely used. It constructs decision trees utilizing a range of samples, averaging them for classification and a majority vote for regression.One of

the most essential characteristics of the Random Forest Algorithm is its adaptability to data sets containing both continuous variables (for use in regression) and categorical variables (for use in classification). The results it produces are preferable for categorization tasks.

Bot-IoT Dataset

The great majority of statistics that are now available are not made for IoT networks and do not include information on recent assaults. Our work focuses on identifying botnets in an Internet of Things environment; hence it needs a dataset with enough data on IoT traces. In the UNSW Canberra Cyber Center's lab, a dataset known as BoT-IoT was created. This dataset combines the label with both the botnet traffic and the regular traffic. In order to simulate various hazardous attacks, the research sets up a huge number of virtual machines on the internal network. Next, the researchers want to gather both legitimate and malicious traffic. To create a dataset for the BoT-IoT, they gather more than 72 million recordings. The dataset includes traffic from a range of malicious attacks, including DDoS, DoS, Information Theft, Keylogging, and DDoS attacks on the protocol that was being used. It also includes further DDoS and DoS attacks that were put up against those attacks. In order to create many diverse botnet situations, the BoT-IoT dataset is realistically built in the IoT network utilizing a range of technologies.

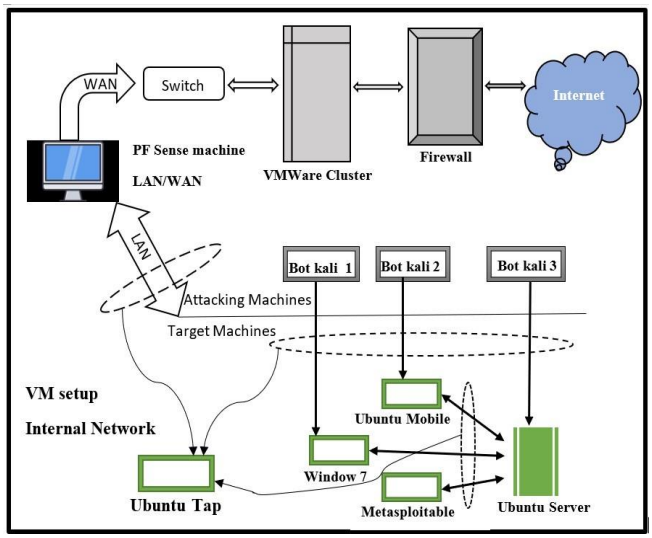

Fig5: Botnet IoT- Dataset

The BoT-IoT dataset comprises traffic collected in groups based on the types of threats and includes a realistic testbed. The 74.csv files that make up the BoT-IoT complete dataset each contain about one million entries, while the dataset as a whole has 72 million records. These data cover traffic from botnets as well as ordinary traffic. Because the system we utilized took a large amount of execution time on our device and had a constrained system capability, we decided to prepare my model using one of the.csv files. There are 999,610 records in this specific file, of which 994,828 are botnet records and the remaining records are regular records. This data set's main distinction is that it contains more than 99% of traffic from botnets but less than 1% of ordinary traffic.

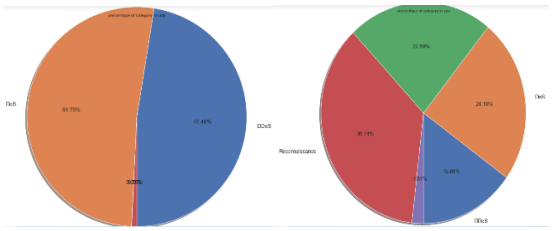

Fig3: Percentage of each attack category for protocol TCP and ICMP

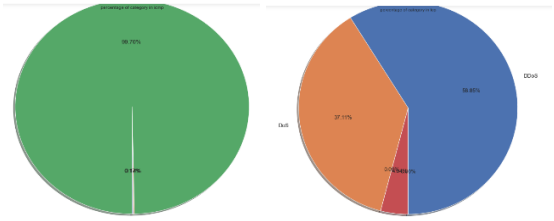

Fig4: Percentage of each attack category for protocol ARP and ipv6-icmp

The Confusion Matrix

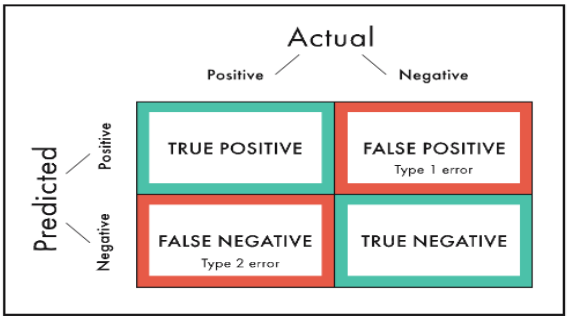

Fig6: Confusion Matrix

Accuracy*:*

The ratio of the number of accurately predicted classes to the total number of predictions makes up accuracy. It is displayed as a percentage of the total. When determining accuracy, True Positives (TPs) and True Negatives (TNs) are considered to be essential.

Precision:

The term "precision" refers to the proportion of accurately predicted positives to the sum of all positives that were forecasted. Precision is evaluated in order to cut down on the number of false positives.

Recall:

The ratio of the number of accurately anticipated positives to the total number of positive cases is known as recall. The analysis of recall aims to reduce the number of false negatives.

F1-Score:

The F-Score metric combines accuracy and recall into a single score value, providing a solution that strikes a balance between the competing priorities of precision and memory. When both false

positives and false negatives are considered to be significant, the F1-Score is evaluated.

Accuracy	$ACC = \frac{TP}{TP+FP}$
Precision	$PPV = \frac{TP}{TP+FP}$
Recall	$TPR = \frac{TP}{TP+FN}$
Fall-out	$FPR = \frac{FP}{FP+TN}$

Fig7: Machine Learning Evolution Metrices

MLA's	Accuracy	F1score	precession	Recall
MNB	0.85671	0.84425	0.88725	0.85671
KNN	0.999979	0.999979	0.999979	0.999979
Random Fore	0.999855	0.99980	0.99983	0.99985
Decision tree	1.0	1.0	1.0	1.0

Fig8: Table representation MLA's Performances

CONCLUSION

In this paper, it suggests that the K-nearest neighbors' method is a good way to find botnets, and we evaluate how well it works. Figure 8 shows how machine learning compares to other well-known techniques and algorithms when it comes to finding and stopping DDoS attacks in IoT networks that are based on botnets. When we compare how to find botnets using real-time unbalanced datasets vs. balanced datasets, we greatly broaden the scope of our investigation. It showed how and why real-time datasets that were not balanced were not good, how this affected metrics like precision, recall, accuracy, and f1-score, and how the dataset could be improved. It also showed how and why the datasets were not good enough. Even though using an uneven dataset showed us that the accuracy was pretty good, the recall and f1-score were both pretty low. This shows that the accuracy we found in the skewed dataset could have been a mirage. Based on the results, the KNN method was found to be the most accurate way to find botnets.

Future Work

Simulating the suggested model in order to evaluate its viability in real time will be the primary focus of any further work related to this study. This concept is capable of being realized through the use of software-defined networks (SDN). On the controller of software-defined networking (SDN), the addition of botnet detection and mitigation measures, which identify the botnet and block the host that is delivering botnet packets across the network, may be utilized. Because of this, the traffic flow in all of the linked hosts will be easier to monitor, and the server's vulnerability to botnet assaults would be significantly reduced.

REFERENCES

1) M. Dibaei, X. Zheng, K. Jiang, R. Abbas, S. Liu, Y. Zhang, Y. Xiang, and S. Yu, "Attacks and defences on intelligent connected vehicles: a survey," *Digit. Commun. Networks*, vol. 6, no. 4, pp. 399–421, 2020.
2) C. D. McDermott, F. Majdani, and A. V. Petrovski, "Botnet Detection in the Internet of Things using Deep Learning Approaches," in *Proc. Int. Jt. Conf. Neural Networks*, vol. 2018-July, Institute of Electrical and Electronics Engineers Inc., oct 2018.
3) Y. N. Soe, P. I. Santosa, and R. Hartanto, "DDoS Attack Detection Based on Simple ANN with SMOTE for IoT Environment," *Proc. 2019 4th Int. Conf. Informatics Comput. ICIC 2019*, pp. 0–4, 2019.
4) M. Aamir and S. M. A. Zaidi, "DDoS attack detection with feature engineering and machine learning: the framework and performance evaluation," *Int. J. Inf. Secur.*, vol. 18, no. 6, pp. 761–785, 2019.
5) Bahsi, S. Nomm, and F. B. La Torre, "Dimensionality Reduction for Machine Learning Based IoT Botnet Detection," *2018 15th Int. Conf. Control. Autom. Robot.*