iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VI **Month of publication:** June 2022

DOI: https://doi.org/10.22214/ijraset.2022.44253

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

A Report on Botnet Detection Techniques for Intrusion Detection Systems

Sathya D[1], Adithi P[2], Cemon Sharon Barboza[3], Bhoomika S [4], Chaitra B Katoti[5]
Dept. of Computer Science and Engineering, Dayananda Sagar College of Engineering, Bangalore

Abstract— A botnet is a malware that degrades the functionality as well as access to a healthy computer system through malware programs. Botnet programs perform DDoS attack, Spam, phishing attacks. Botnet attack takes place in two ways which are peer to peer attacks and command and control attack. The peer-to-peer attack takes place to by passing botnet attacks from one system to another in a peer-to-peer network while the command-and-control attack takes place by a botmaster attack on a server which uses various transactions in exchange with systems on the network and those nodes in the networks function as slaves. The report presents a survey of various techniques of botnet detection models built using several types of machine learning techniques. The report gives the review on various methodologies involved in Botnet Detection and to identify the best methods involved to understand various dataset. We also surveyed on how classification, clustering is used in detection of Botnet to improve the accuracy of the model.

Keywords—Command and Control Botnet, Peer to Peer Botnet, Network Security, Machine learning, Network Protocols, Cyberattacks, Clustering, Classification, Deep Learning

I. INTRODUCTION

The Botnet refers to devices that a hacker control remotely. Botnet is a combined term of robot's interaction with network, where there are two important participants the Botmaster and the Bot slave. The Bot slave acts as a slave of the Botmaster and does what Botmaster asks to do. The Botnet's task is to start attacks by giving instructions from the botmasters to the bot clients to function as slaves to the botmaster. Nowadays botnet attack takes place so silently that the Anti Malware software are not at all able to detect it. The botnet attack taking place in peer-to-peer networks have become a challenge as detecting the centre of command is not so easy. But on a general note, tough it is tough to determine botnet command-and-control attacks, it is possible to observe patterns in data to get a complete picture of network data exchanges and detection of Botmaster is possible.

In DDoS attack, the attacker who is the botmaster has high-end computing Systems and servers to run command & control malware programs which instructs the machines in the next layer or level called handlers. These handlers attack the clients called making them the bot slave. Botnet malicious activities is detected using a variety of methods. As known from the above information the malware detection software finds it incredibly difficult to detect these attacks. The typical approach can be by analysis of network traffic data obtained by simulation and of botnet on Virtual machines and obtaining suitable communication and TCP and UDP protocol network exchange data. The Supervised Learning algorithms (ex: Decision trees, Support Vector machines (SVM)) can is effective in classifying normal traffic from botnet traffic. When Unsupervised learning algorithms like the K-means algorithm integrated with classification algorithms, the outcomes get improved. Multilayer deep learning Neural networks is also a better way to approach large volume of network traffic data. It gives us better chances in identifying other different patterns in data apart from machine learning algorithm. General datasets observed to be used for such type of analysis are: CTU-13 Dataset, KDD cup nineteen dataset, UNSW-NB15 Dataset and Bot-IoT dataset. There are many more datasets developed or used depending on the researcher's goal. The selection of the dataset is a particularly important criterion to undergo a good accuracy, stable model to build a better Botnet Detection system.

The feature selection process is an important aspect of every Machine Learning model. Selecting proper features according to the information needed depends upon the major agenda of the Researcher that aims to detect and analysed. In the event of Command and Control (C&C) botnet attacks, changing IP address data, various times and techniques of TCP data exchange may be a crucial feature to take hold of the Botmaster. In this way majority of the datasets are processed and features are selected as needed. The major step of botnet detection is selection of suitable techniques in which accuracy of the model is improved and the misclassification of data can be prevented. Many researchers have described various methods to detect various kinds of botnets. The different approaches used in different research work is describe further in this paper in other sections.

II. RELATED WORK

The General view of methods using machine learning regarding detection of Botnet Malicious activity is shown below in the Figure 1.

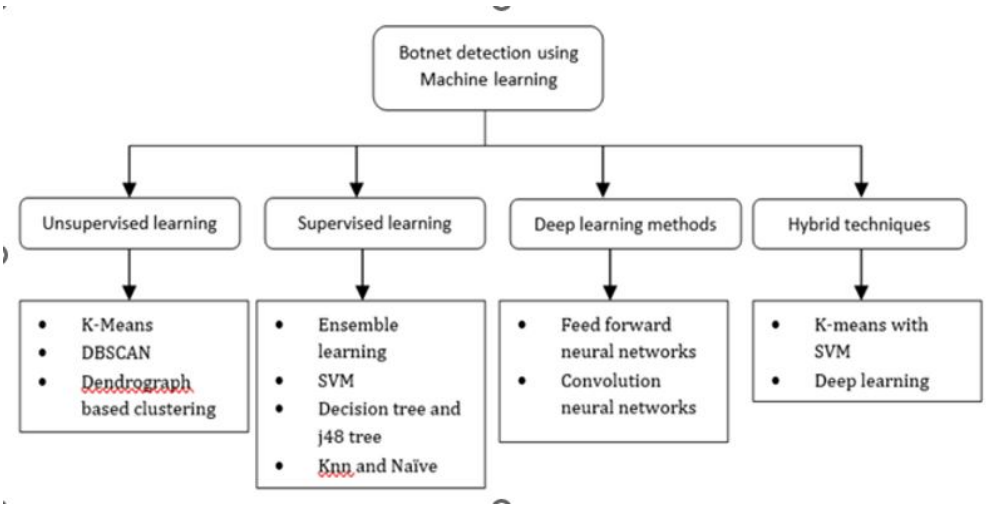


Fig.1 Outline of methodologies used in Botnet Detection Using Machine Learning

The use of BARCA framework [1] has been seen to use feedback-based approach than using historical data, considering metrics correlations, and it is designed to adapt to changes in the behaviour of the system using an incremental and iterative process.
BARCA [1] has three main components which are Behaviour extractor (BE), Behaviour Identifier (BI) and a Feedback Provider (FP). The Behaviour extractor (BE) extracts and collects the performance information of the System periodically and creates a Behaviour Instance whenever new periodic data gets collected. Behaviour Instances are representations of system behaviour obtained timely. The Behaviour Identifier (BI) uses a Behaviour model to classify each Behaviour Instance as normal or abnormal. When the data is classified abnormal, the Behaviour Identifier feedbacks to the Feedback Provider (FP) [1]. The Feedback Provider uses the information from the Behaviour Model to alert the administrator of the system in scan or the user, which acts accordingly to prevent the activity and consistently reports current states of system when alerts are received.
The Behaviour Identifier (BI) uses the combination of single class classifier and multiple binary classifiers to be able to reach the main goals which are: the one class classifier detects the set of anomalies while the various binary classifiers are used to detect anomalies that are known. The Behaviour model is built using Gradient Descent and Support Vector Machines. BARCA [1] uses the user supplied feedback to build its behavioural model. BARCA's advantage is that it does not use Historical Data and hence can dynamically change the system's behaviour accordingly.
The next method found was Twofold approach to build Botnet detection models [2]. The Command-and-Control botnet network (C&C) data capture is the basic method used. Botnet detection is conducted using a new two-fold method The figure 2 shows the Two-fold approach for botnet detection.

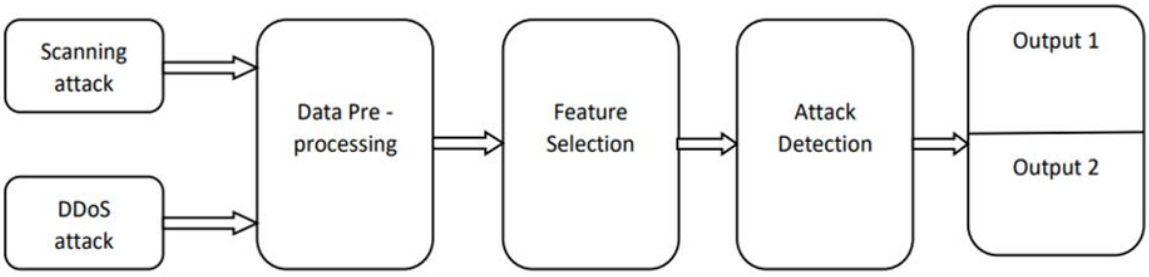


Fig.2 Two Fold Approach for Botnet Detection

The first fold uses the computer vision algorithm ResNet-18 and a state-of-art deep learning algorithm to train a neural network to detect the pre-attacking stages of botnets. This stage is developed to prevent initial attack trails on the IoT botnet network. The Second fold trains a ResNet-18 model for detecting DDoS Attacks, the common type of C&C attack seen as per protocol. To stop and find IoT botnet assaults, the proposed strategy in this research achieved 98.89 % of the accuracy, 99.01 % of the precision, 98.74 %of the recall score and a f1-score of 98.87 %. Overall, this research only covers thirty Data Pre - processing Attack Detection Feature Selection Scanning attack DDoS attack Output 1 Output 2 three types of scans and can detect about sixty types of DDoS attacks. The improvements on upgradation of the model to detects more types of DDoS attacks is yet to be

addresses by the researcher. Also, many distinct types of scans are also needs to be implemented to train the proposed framework.

Many researchers have proposed CNN-based methods. Of them the PCCN method [3] is found to be most efficient and most reliable method in Botnet detection. The Parallel cross convolutional neural network [3] is made up of two parallel convolutional neural networks (CNN) that use feature fusion to work on categories with minor data samples. Because CNN is an image processing and classification algorithm that uses images as input, the data must be converted into a 2D grayscale image for the PCCN model and CNN to use the network traffic data for training.

CNN is chosen as the best algorithm for feature learning to elevate and improve the classification metrics in an imbalanced anomalous dataset. CNN uses the Convolutional operation along with better feature selection provided in Image classification techniques. PCCN [3] improves the classification accuracy of the multi-class and imbalanced network traffic data. For feature learning, the PCCN has two branch networks, the first of which employs fully convolutional networks and the second of which uses standard CNN. The CNN branches' feature fusion makes the network's features more recognizable and robust.

CICIDS2017 is a dataset created in 2017 by the Canadian Network Security Research Institute. The data was extracted with the help of the CICFlowMeter-V3 tool. Many proposed methods have also used The Supervised Learning Techniques which are Random forests [4] and Synthetic Minority Oversampling Technique (SMOTE) [4] to have better feature selection technique on the give network traffic data.

The 'Synthetic Minority Oversampling Technique (SMOTE),' a suggested IDS framework, is used to increase the sensitivity of the classified towards the dataset's minority class data. For feature reduction, the feature selection is done based on the information gain is obtained. The random forests classifier is a part of the framework that is used for classification. The components in the system are said to be executing sequentially. To the desired level, the SMOTE layer will apply oversampling on the data having the minority class. The SMOTE algorithm generates data of training, which can be assumed to be balanced, will be the input for the feature selection. In SMOTE [4] the feature selection part calculates generated data's Information gain (IG). The features will be prioritized based on their IG values obtained after oversampling. The above proposed method gave a better precision of 0.98 on an average with better detection values. The improvements proposed by the researcher is to hybridize the model to be better usable as a real-time Botnet detection system and prevent Botnet attacks.

The next method uses a honeypot [5] designed to be affected by several types of botnets to collect data on their behaviour and how they interact with different systems based on the Architecture and Operating System as well as detects the time needed by a botnet to attack a system. Botnet capturing block (BCB) and Botnet analysis block (BAB) are two blocks in the IoT-BDA architecture technique [5] that capture and analyse IoT botnet samples, respectively. Simulator, payload extractor, and reporter are all part of the proposed honeypot concept. The simulator creates a botnet and lets the attackers to interact with them.

Honeypots can be classified as low, medium, or high interaction depending on the interaction level enabled by the simulators. The "low-interaction honeypots" provide with the only a few allowances providing the services or the protocols they can simulate, with no access to the OS in which the simulation runs. "Medium-interaction honeypots" have more complexity than the lower ones and offer the complete services or protocols they can simulate, and they offer a complete service/protocol implementation. The medium interaction honeypots, on the other hand, do not interact with OS. Honeypots with a lot of interaction have a lot of interaction with the OS. Multiple honeypots imitating different botnet vulnerabilities can be found in the Botnet Capturing Block [5]. The honeypots send their samples automatically to the Botnet analysis block (BAB), which analyses the botnet samples. The behaviour analyser obtains features from the data obtained that may not be seen during static analysis. By detecting IoC and IoA, behavioural analysis can help to detect the intrusion as well as to come out of it. The following method proposed [5] may not have active usage of Machine Learning techniques but uses a different approach of designing honeypots to analyse the Botnet attack.

The next method is used to detect DNS-based botnet [6]. It uses Gunner System [6] approach of Feature Detection using Principal Component Analysis using complex techniques for the analysis. Botnet identification is predicated on the presence of anomalous DNS Response and Querying behaviour. Data pre-processing aids in the production of more accurate results. IGR-Based Features Ranking is used to select DNS features. Various Algorithms that can be used to measure the effectiveness of the process of detecting the features which are Information gain ratio (IGR), Principal component analysis (PCA) and Chi squared techniques. The Information gain ratio measures and to finds by ranking the features based on their effectiveness. The purpose using of feature intersection technique is to reduce the number of relevant characteristics while maintaining accuracy. The detection was made by observing DNS network traffic behaviour. This type of an analysis will be a highly challenged tasks when there are modifications in the botnet acquiring several other techniques which help them bypass the security of the system. This is highly observed in peer-to-peer botnets.

The usage of Externally built machine learning with various modelling tools have also been seen in various aspects. The WEKA tool is a classification and clustering algorithm that is often utilized. The false positive rate (FPR) and true positive rate of

various supervised ML (machine learning) algorithms like the Naive Bayes Classifier, Rule Decision Table, Trees, Ibk Classifier, and J48 Classifier are evaluated to find the most effective one [7]. The simulation picture for K-means clustering using WEKA tool is shown in figure 3.

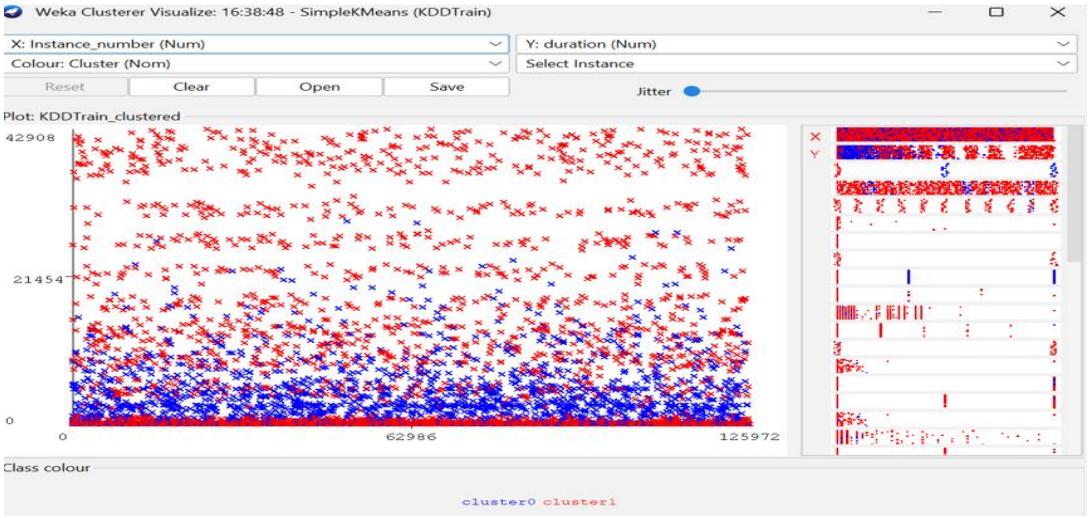


Fig.3 WEKA Tool for Clustering

When classification was done on this dataset The Tree j48 classifier had an elevated level of accuracy. When Clustering was applied on the dataset: The K-means clustering algorithm has been found to be cluster the given data into proper clusters having similar classes as needed. Hence, A hybrid approach is proposed in this research, which is based on the mean percentage of erroneously classified instances by clustering and classification, which is combination both of classification and clustering techniques. The classifier uses Tree J48 algorithm while K-means is used the clustering technique which works on variant dataset partitions. As a part of improvement of this methodology to generate better results, efficient data partition techniques to be applied as specified by the researchers.

The proposed solution is an enhanced Peer Hunter [8], a community behaviour analysis-based network-flow level system which is capable to detect peer to peer(P2P) botnet in the waiting stage. The Command-and-control botnet channel shall be encrypted and botnet activity on the same site overlaps with legitimate P2P traffic. There will be no patterns in the traffic data known in advance (unsupervised or clustering). Using "mutual contacts," the bots are subsequently organized into communities. Finally, it employs community behaviour analysis at the network-flow level to detect any potential botnets.

A study of P2P network flow detection with various DDs (destination diversity) produced rate of detection on average and false positives with DDs in the range of 2- 13,500. In this study, different MCRs (mutual contacts ratio) were used for evaluating community detection performance. The results expressed that the system is quite successive and robust at identifying bots versus legal hosts, in addition to distinguishing between distinct types of bots. In comparison to most P2P botnet network flow cluster communities, legal P2P network flow communities had a lower AVGDDR (average destination diversity ratio), which was determined by evaluating various parameter values for the botnet identification component. An optimal set of parameter values was used to get a detection rate of 100 percent with no false positives [8].

The effectiveness of the botnet detection technique that uses DNS query data analysed using a range of machine learning algorithms, which includes decision trees, kNN, random forests, and Naive Bayes [9,17,22]. The detection model is developed in two steps: training and detection. As part of the training phase, DNS query data and domain names within the queries are collected. Training features are extracted from a pre-processed list of domain names. Machine learning methods are used in the training phase to learn classifiers. The machine learning algorithm resulting in the best classification accuracy will be used in the detection model after the evaluation phase. In the phase of detection, DNS queries are monitored and processed. Domain names are extracted and pre-processed to determine whether a domain name is valid or a botnet domain for which the system makes use of the classifier that was built during the training step. Experiments showed that the random forest approach far outperformed C4.5 decision tree algorithms. To detect botnets, the random forest machine learning method is used because of its high accuracy in classification [9].

Similar Techniques of using different supervised techniques with CTU-13 dataset and UNSW-NB15 datasets are also proposed [17,22]. Use of Logistic Regression in detection of Botnet using various type of numerical datasets are proposed which have given the better accuracy [18]. Ensemble learning have gained popularity over years in many diverse types of research. Random

Forests, XGBoost, Cat boost have been widely used, and have also been applied in Botnet Detection on GTCS dataset and found to be dependable with basic accuracy of more than 90% in majority of algorithm's ensemble [19]. Among this J48 ensemble has shown remarkably high accuracy [19].

Feed forward Neural network-based algorithm is also established in various research [21]. Multilayer feed-forward networks are used to solve situations in which all the data must be supplied to the network of Multilayer feed-forward networks parallelly. During the phase of training, a training set of data 80% of the obtained normalized dataset is provided as the input to the Neural net, is iterated and adjusted with proper parameters such as the bias and the weight to obtain an output of high accuracy [21].

A novel approach was taken to create a framework that relies on the possibilities inherent in PF RING for sniffing and analysing network traces to dynamically extract flow features, extract promising dialogue features using a random forest model, and evaluate the effectiveness of various categorization techniques. This method is described using the well-known CTU13 dataset and non-malicious software [10]. The framework includes: a traffic processing module that groups captured packets that are placed in multiple flow buffers, a flow-based feature extraction module that creates statistical flow characteristics, a conversation-based feature selection module that extracts promising conversation-based feature sets, and a botnet detection module that uses a machine Learning algorithm to identify network traffic that could potentially be a botnet attack. The CTU was used to examine the performance of the five best classifiers, namely Random Forest, REPTree, Bayes Net and DecisionTump [10], using botnet and regular traffic. Botnet traffic can thus be identified in a high-speed network environment using a semi-real-time botnet detection platform based on random forest classifiers and conversational capabilities. The advantages of existing identification methods based on the statistical behaviour of the flow and the similarity of the flow are combined in conversational features. In the future, according to the proposed conversational features, the researchers want to focus on the regulations of the mining associations [10].

A new graph-based approach has been used also called as SOM algorithm [20]. The SOM algorithm uses a graph-based clustering technique where each node is connected by directed connecting edges. Support Vector Machines' power is confined to a single set of features in a single dataset. however, this SOM technique performs well even with datasets with distinct characteristics [20].

The different approach considered here is network traffic behaviour analysis and machine learning to characterize network traffic behaviour to detect botnet activity. There are two steps to the detection framework: Detectors are provided a set of known harmful and non-malicious data attribute vectors during the training phase to train classifiers in the detection of the two types of data. The system then moves on to the detection phase, where it monitors network traffic and classifies the attribute vectors created by active flows. The features were chosen based on Storm, Nugache, and Waledac [11], which are well-known protocols and botnets. With the Reduced Error Pruning technique (REPTree), better detection accuracy is obtained even when inputs are noisy, and the tree size is reduced to reduce classification complexity. The WEKA tool's ML (Machine Learning) libraries built in Java are used [11 In the decision tree classifier, over 90% of predictions were correct, and false positives were low. Based on the network traffic characteristics monitored at specific intervals, the model detects bot activity during both the C&C and assault phases. In this case, it detected botnet attacks with high accuracy using a decision tree classifier. Moreover, it would be impossible to install and operate a single detector on the network devices or on a network with more than one hundred nodes. To detect threats in vast network environments, planned is both resilient, scalable, and able to handle a wide range of events.

This novel approach proposed examines current heuristic malware detection approaches and provides a brief review of major elements employed in these methods, such as API Calls, Opcodes, and N-Grams, as well as their benefits and drawbacks [12]. To study the behaviour of an executable file, heuristic methods employ data mining and machine learning techniques. Features used for heuristic methods are API calls, CFG, N Gram, Opcode, and hybrid features.

All the programs use API to send calls and requests to the underlying OS. It beats standard antivirus software like McAfee Virus Scan and Norton Antivirus, as well as prior data-mining-based detection methods [12] in terms of performance, efficiency, and low time complexity. An Opcode is a subset of a machine language command that specifies the action to be taken (short for Operational Code). It can detect malware that has been disguised. Imbalance datasets are well-fitting and help to limit the frequency of false positives. N-Grams are all substrings of a bigger N-gram string. Tesauro et al [12], are the first in attempting to use N-Grams where they used Artificial Neural Networks and NGrams to detect Boot Sector Viruses. It effectively improves malware detection accuracy. The Control Flow Graph (CFG) is the graph which depicts program control flow, is frequently used in the software analysis. In hybrid features, there are two key elements that determine the success of machine learning classifiers: features and algorithms. Some researchers wanted to use features to increase the precision of machine learning. To improve accuracy, they mix features [12].

A unique Quality of service (QoS) based classifier for Virtual private network (VPN) based on PHB which is the per-hop-behaviour is the given domain is presented as a solution [13]. The proposed QoS approach concentrated on the initial steps of QoS managing, Traffic Classification and Marking.

A network's Traffic Classification stage determines the type of applications and protocols that exist in the network. In the Traffic Marking stage, service policies are applied to QoS fields on packets incoming and outgoing. A2 dataset identifies PHB-Labels for non-VPN traffic ranging from AF12 to AF32, while those for VPN traffic range from AF11 to AF31. The B dataset contains all PHB labels from AF11 to AF32 and their priority numbers [13]. In a cross-validation test, the C4.5 method performed slightly better than kNN when using a single run for cross-validation rather than a T-test for traffic labels. Bagging and boosting performed better than C4.5 in the Traffic-Labels technique, which employed only two machine learning algorithms (i.e., kNN and C4.5). For dataset B for Bagging at 15 seconds ftm, the accuracy achieved with PHB Labels is 86.94 percent. For all ftm values, kNN, Naive Bayes, and Multilayer Perceptron perform poorer than C4.5, rules, Bagging, and Boosting Future research could result in the creation of a domain ontology for recommender system concepts. Ontologies enable a network traffic labelling recommendation system that is captured with the collection of time-related information [13].

This new study focuses on threats to the network and lays out core architectural components for creating awareness of the situation across various enterprise, cloud, and fog or edge deployments. The main limitations are enterprise: Limitations on Portable devices, IoT, Public internets such as Wi-fi, and their weak security measures [14]. A framework for identification of network threats contains: the programmable agent which is used for traffic inspection, Action Enforcement, and simple analysis. Local controller manages all the agents that are programable by discovering the resources along with their topology, translating technology-agnostic programs and configurations into device protocols that are specific, and collecting and aggregating measurements, events and the data obtained. The orchestrator distinguishes between centralized and distributed detection strategies in computing operations using abstractions and models. The legal repository oversees storing data and events for cybercrime investigations, and offline analysis and evidence of court in a safe and dependable manner. The human interface is a tool interactive with outside world, which shows the current condition of cybersecurity and allows for quick and intuitive reactions to attacks [14]. As a result, the framework demonstrated the fundamental components of a distributed architecture that separates context management from the detection logic, allowing for greater flexible security appliance deployment and upgrade. This research could serve as a model for building new cyber-security paradigms, driving parallel but complementary research and development [14].

The strategy of using Machine Learning to give a framework for modelling attacker behaviour: Within the same botnet, attackers in principle conduct their attacks near each other. Next, the use of a specific form of factor system, known as the Multivariate Hawkes Process, explicitly assault the notion of temporal styles [15]. Among the contributions is the creation of a tool that models and organizes attacker behaviour only based on temporal data, eliminating the need for time-consuming feature engineering. Attacker clustering can be divided into two parts: latent shape detection and graph-based fully clustering. The Bayesian Probabilistic Graphical Model (BPGM) [15] and a quality based completely grouping strategy in Machine Learning are used to overcome these issues. To understand illustration and inference encompassing uncertainty, BPGM is utilized to present probability-based relationships amongst random variables by means of a graph. Think of the assault temporal information as a pattern within a Poisson system and determine how comparable those patterns are using the Multivariate Hawkes Process [15].

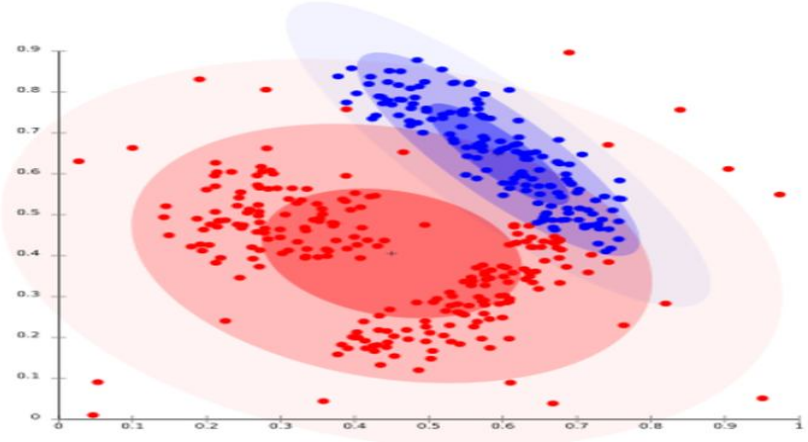


Fig.4 Behaviour Based Clustering

BPGM combines these mathematical strategies. Thus, in contrast to prior attempts primarily based on classifiers and clustering algorithms, the framework can ease the burden of acquiring botnet features. Predictive log likelihood is utilized as a metric. This is the most common method for assessing attacker styles without having to go through the time-consuming process of characteristic engineering. Some shortcomings remain, which should be eliminated in future work [15]. One of the downsides is that if an attacker launches an attack asynchronously, the accuracy of the attack simulation can be reduced, requiring the use of current technical tools to create multiple methods for mixing detected clusters.

Particle Swarm Optimization (PSO) and Kmeans algorithms are used to solve botnet problems and develop a step-by-step botnet identification system. The research does not propose creating a detection system akin to antivirus software but identifying the bot clients based on abnormal network activity. Raw data layer, Data Process layer, Lan IP layer, Behaviour layer, and Cluster layer are the five levels that make up the Botnet detection technique [16]. The Raw Data Layer collects network traffic data from several network devices and stores it in a central database. The Data Process Layer is responsible for converting network traffic data into related information. The information contained in the Lan IP Layer is used to extract IP features for auditing purposes. A similar set of IP properties are merged into a single network behaviour, which can then be divided into three types: Act Behaviour, Fail Behaviour, and Scan Behaviour. Finally, using the PSO algorithm (and therefore K-means clustering) it is found that aberrant client bots (i.e., Bot clients) are often found within the cluster layer [16]. Evaluation of the flow records from the core switch involves sending them to a server for data flows supported by the Cisco NetFlow Protocol and converting them into the database. In this solution, a simple and practical method was proposed to detect Botnet behaviour using network behaviours, such as Act Behaviour, Fail Behaviour, and Scan Behaviour as well as pinpointing the Bot client using PSO+K-means clustering [16]. Infected botnet client hosts are typically easily identifiable. A dormitory network or a home network, as well as mobile 3G networks, frequently use it [16]. Figure 4 shows the behaviour-based clustering.

Many hybrid methodologies are also specified by various researchers which propose the usage of clustering and graph clustering techniques in Botnet detection. Many P2P Botnets show high detection ability in this technique being used. The references [28,29,30] show significant research on usage of detection of anomaly in the system using community based and centrality-based approaches. These approaches show how the enormous number of Botnets can be detected using simple Unsupervised learning approaches. Many of these approaches either use self-created datasets, by designing honeypots or using attack on system.

Various other researchers suggest using Clustering and classification techniques to conduct Botnet detection in Networks. The table I suggests five such research which can be observed to know various research methodologies.

TABLE I
ANALYSIS OF VARIOUS BOTNET DETECTION METHODOLOGIES

Paper Name	**Related Work**		
	Methodology used	**Dataset Used**	**Features and Enchancements**
Investigation of peer-to-peer botnet using TCP control packets [23]	Data extraction and Filtering is done on the needs specified and Decision tree is applied to the filtered data	IOST dataset	• Peer to Peer Botnet Detection is Successful • Improvement of Accuracy and performance expected
Botnet detection using Artificial Intelligence [25]	Both Clustering and Classification are used. Both the techniques accuracy is observed, and the Machine learning model is built for each type of model and combined to form a detection system	Self-Created using different datasets or Honeypots	• C&C, IRC and P2P detection using multiple machine learning models • Using General behavioral patterns more than Signature based patterns to

			be conducted
Bot Revealer [24]	Pattern in Botnet traffic detected. Network flow data extracted and analyzed; Behavior based approach instead of Machine Learning used	Self-Created dataset	• C&C Botnet Detection done • Using General behavioral patterns more than Signature based patterns to be conducted
Naive Bayes Classifier with Feature Reduction for Intrusion Detection [26]	Uses WEKA tool to do the classification analysis on the chosen dataset. Forty-one features of the dataset selected as per the domain specificity and accuracy	NSL-KDD dataset	• C&C Botnet Detection • Customizing Feature Selection
Botnet Detection via network traffic flow [27].	Uses various possible Supervised. Unsupervised Learning techniques such as K-Means,	CTU-13 dataset	• Both C&C and P2P botnet detection • Developing a layered design that works with high speed.

The rapid-fire advancement of IoT technology has concentrated experimenters ' and technologists ' attention on the design of IoT healthcare systems. numerous IoT healthcare systems have been proposed in recent times, but these systems endure the security backdoor. IoT healthcare systems ' security is pivotal as any security breach orcyber-attack in similar systems may beget a rigorous effect on mortal life and indeed may beget death in some cases. thus, in this work, we proposed a frame for developing IoT environment- apprehensive security results to descry vicious business in IoT healthcare surroundings. The proposed frame is composed of an IoT business creator tool in which an IoT- grounded ICU use case is created to induce standard and vicious business. The generated business is also converted into a dataset by rooting the features using a python script. latterly, we trained and tested six generally used machine literacy(ML) classifiers over the generated dataset for vicious and traditional business discovery in the IoT healthcare terrain. Eventually, the data is tested and analysed the performance of each trained ML classifier. Among the six ML classifiers, the Random Forest classifier performed the stylish with99.7068 perfection,99.7952 recall,99.5123 delicacy, and 99.6535 F1- score. The experimental results demonstrate the effectiveness of the proposed frame for developing effective IoT environment- apprehensive security results. also, the proposed frame and generated dataset are helpful for the experimenters to pursue the proposed system for developing further robust environment-apprehensive security results, especially for IoT healthcare surroundings. likewise, with the proposed frame's help, the experimenters can snappily induce the business of other IoT use cases in order to develop AI- grounded security results for other IoT use cases[30].

With the nonstop relinquishment of the IoT paradigm in critical structure and consumer sectors, their security and sequestration enterprises are getting relatively serious, leading to ruinous consequences. The following research respects current IoT- centric exploration by offering a macroscopic, general and unresistant methodology to infer Internet- scale compromised IoT bias and to report on ongoing IoT botnets. The work originally introduces a new darknet- specific sanitization model that contributes to the field of Internet measures at large. latterly, by contriving a double classifier grounded upon a CNN in confluence with active measures, the proposed work is able of fingerprinting compromised IoT bias by solely operating on darknet traf- fic. Accordingly, by automating the generation of autographs related to the anchorages being probed coupled with their distribution in addition to other simplistic yet effective features, the proposed approach provides the capability to infer ongoing orchestrated botnets. The results demonstrate the significant security issue with the IoT paradigm by exposing further than exploited IoT bias

during only a 24- hour period, some of which have been stationed in critical sectors similar medical and manufacturing. also, the outgrowth provides substantiation- grounded pointers affiliated to ongoing IoT botnets similar as those of Mirai, Hide and Seek, and Reaper, to name a many. More interestingly, the results demonstrate evolving IoT botnets with crypto jacking capa- bilities, where numerous of those feel to be attributed to the same architect by exposing the same employed key[31].

The use of damped incremental statistics and the Z-Score method to extract and normalise the 23-dimensional basic features of inbound and outbound traffic (including benign traffic and five kinds of attack traffic) of IoT devices. Then we apply the TAM-based MCA algorithm to correlate the original 1 × 23 dimensional features and generate 23 × 23 dimensional feature matrices, and each matrix can be regarded as a grayscale image, and the grayscale images generated from different types of traffic have obvious difference. Based on this premise, we design a convolutional neural network to learn the MCA matrixes, and the final experiments achieve satisfactory results. Therefore, the approach with preprocessing and CNN proposed in this paper have a good performance in IoT botnet detection[32].

The main donation of the proposed system is to descry IoT botnet attacks launched form compromised IoT bias by exploiting the effectiveness of a recent mass intelligence algorithm called Grey Wolf Optimization algorithm(GWO) to optimise the hyperparameters of the OCSVM and at the same time to find the features that stylish describe the IoT botnet problem. To prove the effectiveness of the proposed system, its performance is estimated using typical anomaly discovery evaluation measures over a new interpretation of a real standard dataset. The experimental results show that the proposed system outperforms all other algorithms in terms of true positive rate, false positive rate, and G- mean for all IoT device types. Also, it achieves the smallest discovery time, while significantly reducing the number of named features [33].

The network terrain is designed and enforced using Network Simulator. The simulator executes the proposed dragonfly clustering protocol for cluster head selection and cluster conformation to reduce the cluster breakage and to ameliorate the reading effectiveness in the network. In the considered 100 bumps in the network, there are three types of detectors similar compendiums, markers and cluster heads. All the compendiums are homogeneous in the cluster but perform different tasks. This type of distribution balances the functional cargo within each cluster and also results in bettered network continuance. The cluster head schedules data collection time in the network. compendiums smell the data from markers and shoot them to the cluster head within the cluster. Cluster head performs aggregation of the gathered data before transmitting them to the base station. Our simulation result is estimated in terms of number of parameters similar as network continuance(number of Active bumps) and cluster head selection rounds[34].

The number of network security incidents has increased significantly in recent times. Despite the fact that there have been multitudinous botnet discovery studies, the development of farther exploration on the SDN security system is extremely low, as can be easily seen by the lack of current exploration. This specifically pertains to the aspects of defending networks and relating the variations in the attacks with regard to SDN network security. also, to the stylish of the author's knowledge, there seems to be a significant lack of thorough and methodological analysis of the state of the art regarding colorful approaches to botnet discovery. More specifically, this pertains to botnet discovery grounded on ML and botnet discovery grounded on ML in SDN. likewise, it can be easily seen that numerous issues that need further exploration regarding the SDN security system, explicitly in botnet attacks, take this check paper into consideration. This check has reviewed colorful ways for detecting botnets in traditional networks as well as SDN. To overcome all limitations mentioned in this check, we believe graph- grounded features grounded on ML for bot discovery in SDN is the most promising avenue for unborn study. Graph- grounded features, deduced from inflow- position information, which reflect the true structure of dispatches, relations, and geste of hosts is an volition that overcomes these constraints[35].

III. CONCLUSIONS

The survey is conducted on various research articles to understand the approaches used to conduct detection of Botnets using machine learning. The Supervised learning like SVM and Random forests and many more are surveyed. Many methods used clustering like K-Means clustering algorithm and more, while a few also combined supervised learning with clustering algorithms. Further approaches use Deep learning in which CNN and ResNet-18 based approaches was found to be highly accurate and more interesting research. Most of the models uses prebuilt tools such as WEKA tool to obtain their modelling.

REFERENCES

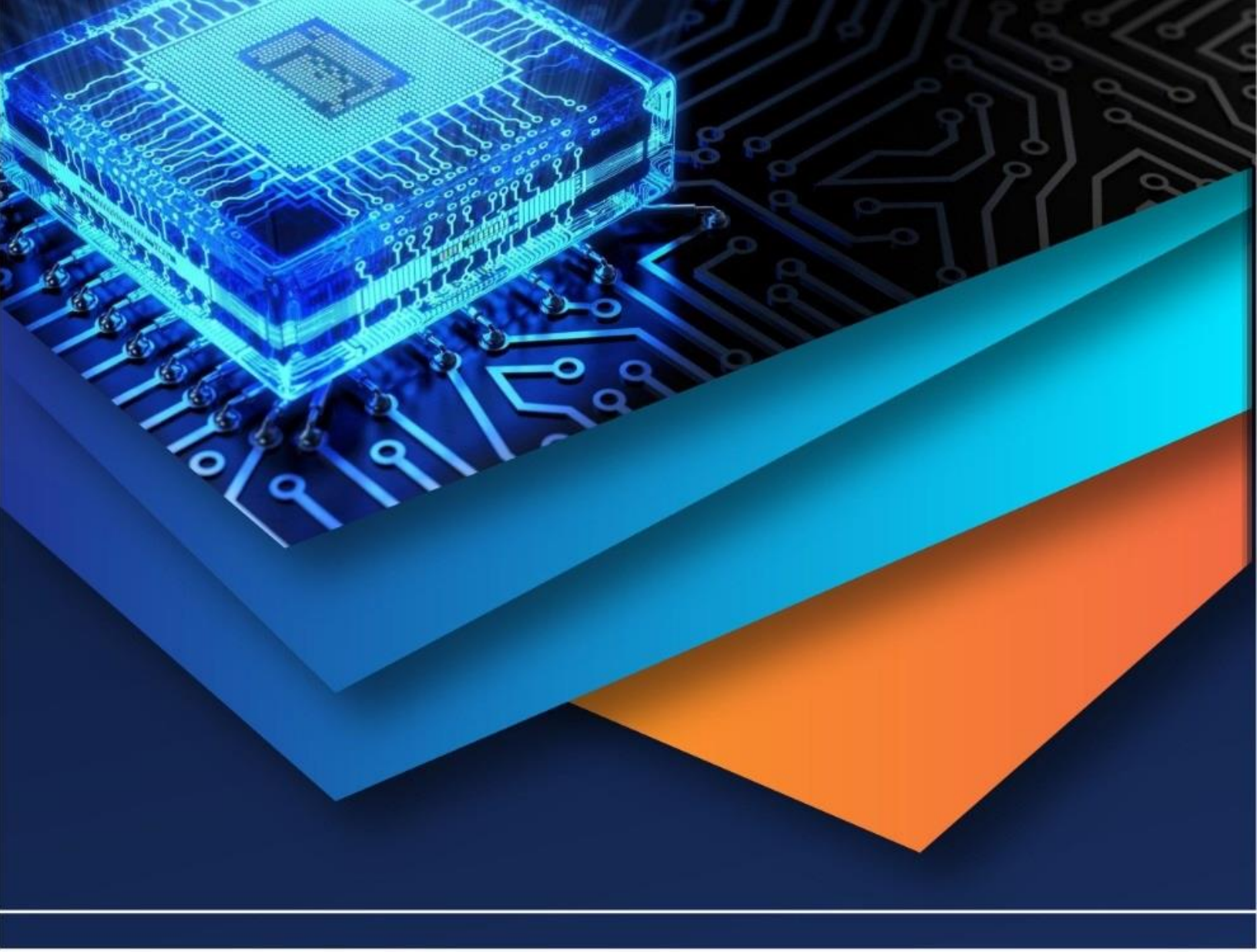
[1] J. A. Cid-Fuentes, C. Szabo, and K. Falkner, ''Adaptive performance anomaly detection in distributed systems using online SVMs,'' IEEE Trans. Dependable Secure Computer., vol. 17, no. 5, pp. 928–941, Sep./Oct. 2018

[2] F. Hussain et al., "A Two-Fold Machine Learning Approach to Prevent and Detect IoT Botnet Attacks," in IEEE Access, vol. 9, pp. 163412-163430, 2021, doi: 10.1109/ACCESS.2021.3131014.

[3] Y. Zhang, X. Chen, D. Guo, M. Song, Y. Teng, and X. Wang, ''PCCN: Parallel cross convolutional neural network for abnormal network traffic flows detection in multiclass imbalanced network traffic flows,'' IEEE Access, vol. 7, pp. 119904–119916, 2019.

[4] A. Esfahan and D. L. Bhaskari, ''Intrusion detection using random forests classifier with SMOTE and feature reduction,'' in Proc. Int. Conf. Cloud Ubiquitous Computer. Emerg. Technol., Nov. 2013, pp. 127–13

[5] T. Trajanovski and N. Zhang, "An Automated and Comprehensive Framework for IoT Botnet Detection and Analysis (IoT-BDA)," in IEEE Access, vol. 9, pp. 124360- 124383, 2021, doi: 10.1109/ACCESS.2021.3110188.
[6] K. Alieyan, A. ALmomani, A. Manasrah, and M. M. Kadhum, ''A survey of botnet detection based on DNS,'' Neural Comput. Appl., vol. 28, no. 7, pp. 1541–1558, Jul. 2017
[7] S. Haq and Y. Singh, "Botnet Detection using Machine Learning," 2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC), 2018, pp. 240-245, doi: 10.1109/PDGC.2018.8745912
[8] D. Zhuang and J. M. Chang, ''Enhanced PeerHunter: Detecting peer-to-peer botnets through network-flow level community behaviour analysis,'' IEEE Trans. Inf. Forensics Security, vol. 14, no. 6, pp. 1485–1500, Jun. 2019
[9] X. D. Hoang, ''Botnet detection based on machine learning techniques using DNS query data,'' Future Internet, vol. 10, no. 5, pp. 1–11, 2018.
[10] R. Chen, W. Niu, X. Zhang, Z. Zhuo, and F. Lv, ''An effective conversation-based botnet detection method,'' Math. Problems Eng., vol. 2017, pp. 1–9, Apr. 2017.
[11] D. Zhao, I. Traore, B. Sayed, W. Lu, S. Saad, A. Ghorbani, and D. Garant, ''Botnet detection based on traffic behavior analysis and flow intervals,'' Comput. Secur., vol. 39, pp. 2–16, Nov. 2013.
[12] Z. Bazrafshan, H. Hashemi, S. M. H. Fard, and A. Hamzeh, ''A survey on heuristic malware detection techniques,'' in Proc. 5th Conf. Inf. Knowl. Technol., May 2013, pp. 113–120.
[13] J. A. Caicedo-Muñoz, A. L. Espino, J. C. Corrales, and A. Rendón, ''QoSclassifier for VPN and non-VPN traffic based on time-related features,'' Comput. Netw., vol. 144, pp. 271–279, Oct. 2018.
[14] R. Rapuzzi and M. Repetto, ''Building situational awareness for network threats in fog/edge computing: Emerging paradigms beyond the security perimeter model,'' Future Gener. Comput. Syst., vol. 85, pp. 235–249, Aug. 2018
[15] P. Sun, J. Li, M. Z. A. Bhuiyan, L. Wang, and B. Li, ''Modeling and clustering attacker activities in IoT through machine learning techniques,'' Inf. Sci., vol. 479, pp. 456–471, Apr. 2019.
[16] S.-H. Li, Y.-C. Kao, Z.-C. Zhang, Y.-P. Chuang, and D. C. Yen, ''A network behavior-based botnet detection mechanism using PSO and K-means,'' ACM Trans. Manage. Inf. Syst., vol. 6, no. 1, pp. 1–30, Apr. 2015.
[17] Mustafa Alshamkhany, Wisam Alshamkhany, Mohamed Mansour, Mueez Khan, Salam Dhou, Fadi Aloul, "Botnet attack detection using machine learning", in Research Gate, Doi: 10.1109/IIT50501.2020.9299061
[18] Mr. A. Sankaran, A. Krithika Bavani Murat, M. Tharrshinee, G. Yuvasree, "BOTNET DETECTION USING MACHINE LEARNING", in International Research Journal of Engineering and Technology (IRJET), Volume: 07 Issue: 07 | July 2020
[19] Ahmed Mahfouz, Abdullah Abuhussein, Deepak Venugopal and Sajjan Shiva, "Ensemble Classifiers for Network Intrusion Detection Using a Novel Network Attack Dataset", doi:26 October 2020
[20] Sudipta Chowdhury, Mojtaba Khanzadeh, Ravi Akula, Fangyan Zhang, Song Zhang, Hugh Medal, Mohammad Marufuzzaman and Linkan Bian, "Botnet detection using graph-based feature clustering", DOI 10.1186/s40537-017-0074-7
[21] G. Kirubavathi Venkatesh and R. Anitha Nadarajan, "HTTP Botnet Detection Using Adaptive Learning Rate Multilayer Feed-Forward Neural Network", https://doi.org/10.1007/978-3-642-30955-7_5
[22] Siqiang Hao, Di Liu, Simone Baldi, Wenwu Yu, "Unsupervised detection of botnet activities using frequent pattern tree mining", doi: https://doi.org/10.1007/s40747-021- 00281-5
[23] Investigation of peer-to-peer botnet using TCP control packets and data mining techniques Mohammad Alauthaman*, Nauman Aslam, and M.A. Hossain
[24] BotRevealer: Behavioral Detection of Botnets based on Botnet Life-cycle Ehsan Khoshhalpour 1, and Hamid Reza Shahriari 1
[25] Botnet Detection Using Artificial Intelligence Astha Parihar1 * and Prof. Neeraj Bhargava2
[26] Intrusion Detection using Naive Bayes Classifier with Feature Reduction Dr. Saurabh Mukherjee, Neelam Sharma
[27] Botnet Detection via mining of network traffic flow Lakshya Mathur, Mayank Raheja, Prachi Ahlawata
[28] Collaborative Botnet Detection with Partial Communication Graph Information Harshvardhan P. Joshi, Matthew Bennison and Rudra Dutta Department of Computer Science, North Carolina State University Raleigh, NC 27695-8206, USA
[29] BotMiner: Clustering Analysis of Network Traffic for Protocol- and StructureIndependent Botnet Detection Guofei Gu†, Roberto Perdisci‡, Junjie Zhang†, and Wenke Lee† † College of Computing, Georgia Institute of Technology ‡ Damballa, Inc. Atlanta, GA 30308, USA [30] BotTracer: Execution-Based Bot-Like Malware Detection Lei Liu1, Songqing Chen1, Guanhua Yan2, and Zhao Zhang3
[30] F. Hussain et al., "A Framework for Malicious Traffic Detection in IoT Healthcare Environment," Sensors, vol. 21, no. 9, p. 3025, Apr. 2021, doi: 10.3390/s21093025.
[31] Morteza Safaei Pour, Antonio Mangino, Kurt Friday, Matthias Rathbun, Elias Bou-Harb, Farkhund Iqbal, Sagar Samtani, Jorge Crichigno, Nasir Ghani, "On data-driven curation, learning, and analysis for inferring evolving internet-of-Things (IoT) botnets in the wild," Computers & Security, Volume 91,2020,101707,ISSN 0167-4048, https://doi.org/10.1016/j.cose.2019.101707
[32] J. Liu, S. Liu and S. Zhang, "Detection of IoT Botnet Based on Deep Learning," 2019 Chinese Control Conference (CCC), 2019, pp. 8381-8385, doi: 10.23919/ChiCC.2019.8866088.
[33] Al Shorman, A., Faris, H. & Aljarah, I. Unsupervised intelligent system based on one class support vector machine and Grey Wolf optimization for IoT botnet detection. J Ambient Intell Human Comput **11,** 2809–2825 (2020). https://doi.org/10.1007/s12652-019-01387-y
[34] Singh Rathore *, Abhishek Kuma2 , Vicente García-Díaz ,Department of CSE, ACERC Ajmer (India), Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab (India) , Department of Computer Science, Universidad de Oviedo (Spain), "A Holistic Methodology for Improved RFID Network Lifetime by Advanced Cluster Head Selection using Dragonfly Algorithm Pramod**",** International Journal of Interactive Multimedia and Artificial Intelligence,2020, 10.9781/ijimai.2020.05.003
[35] K. Shinan, K. Alsubhi, A. Alzahrani, and M. U. Ashraf, "Machine Learning-Based Botnet Detection in Software-Defined Network: A Systematic Review," Symmetry, vol. 13, no. 5, p. 866, May 2021, doi: 10.3390/sym13050866.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY