BotNet Detection for Network Traffic using Ensemble Machine Learning Method

Yogita Barse, Deepak Agrawal

Abstract*: In todays era the need of security is raising due to hike in security risks discovered every day. A new vulnerability can be found in any software or product by the attacker as it launches in the market. Botnet carried out various attacks in distributed manner which results in extensive disruption of network activity through information and identity theft, email spamming, click fraud DDoS (Distributed Denial of Service) attacks, virtual deceit and distributed resource usage for cryptocurrency mining.The main aim f botnet is to steal private data of clients,sendind spam and viruses and DOS attacks in the network. The detection of Botnet like Rbot ,Virut and Neris are still vigorous research area due to unavailability of any technique to detect the entire ecosystem of botnet. As they are comprised of different configurations and profoundly armored by malwares writers to dodge detection systems by utilizing complicated dodging techniques. Hence only solution is to discover the infected botnets to control over the services and ports. This work aims to contribute in the botnet detection with its overview and existing methods.

The study focuses on techniques like one-hot encoding and variance thresholding. These techniques are utilized to clean the botnet dataset. The performance of the machine learning model can be improved with feature selection methods. The work explores the dataset imbalance problem with the help of ensemble machine learning techniques. The performance is evaluated on the best received model that is trained and tested on datasets of various attacks.

***Keywords*: Botnet Detection, Machine Learning, Network Traffic, Security, XGBoost**

I. INTRODUCTION

The fast expansion of mobile technology and mobile computing fields became a sturdy effect in the development of both wireless communications technology (hardware) and mobile-based apps (software). These technologies provide full access to all cloud services all the way through the cloud anytime anywhere at any network. Vulnerability or Development is exposed and exploited by the malicious users or attackers for several purposes [1,2]. Hence security solutions for such technologies and networks are in focus for many research studies. Various distributed attacks are carried out through botnets resulting in network failure or disruption in network activity. Some of them are email spamming, information and identity theft, click fraud, virtual deceit, DDoS (Distributed Denial of Service) attacks and distributed resource usage for cryptocurrency mining. These all malicious activities are carried out with a botnet.

Revised Manuscript Received on November 01, 2020.
***** Correspondence Author
Yogita Barse, Department of Computer Science, Indore Institute of Science & Technology, Indore (M.P.)-India
Deepak Agrawal, Department of Computer Science, Indore Institute of Science & Technology, Indore (M.P.)-India

A. Botnet: Hackers control the malware infected group of internet connected devices which is called a botnet. Botnet attacks are instigated by cyber criminals by utilizing the botnets that incorporate malicious activities such as, unauthorized access, attacks, credentials leaks, data theft and DDoS. These compromised computers are known as bots or zombies. Command and Control server (C&C) is used to control bots through a botherder. The purpose of C&C server is to discover the system vulnerabilities and to perform an attack through commands controls and code updates which effects in various attacks like DDos, information theft, phishing etc. Due to the intrinsic behavior of changing attacks the anticipation or detection is complex [2]. Several research and methods are proposed for the botnet detection and block of botnet still there is a strong need of an effective method which rectifies the security issue of various types of networks. Owing to a thriving need of botnet detection area this research work will aim to counter some research questions arises during the study of existing work done in detection of botnets. Hence the objectives of the proposed work are-

- Length the existing botnet detection methods
- To design a methodology with efficient and accurate training
- Rectify data imbalance of botnet traffic
- Existence of any machine learning model

Hence there is a huge gap exists in identification of all types of botnet attack therefore our aim is to develop a model that provides a solution in scalable detection of botnet attacks.

B. Types of Botnet attacks : Botnet lifecycle consist of five phases as explored in [1, 2, 3].The initialization is the infectious phase in which C&C server scans the network and susceptibility in servers and systems. Then in the second phase a shell script is run on the identified weak system and termed as second infectious stage. These shell script make possible the download of malwares or bot codes from C&C server. When the malware reached to the host system a connection is established between server and botherder and the commands can be initiated through these botherder to the system. These commands are malicious and interrupt the network services. An update and maintenance phase is also required as a perpetual to avoid the detection. Some of the known methods of botnet-

Denial of Service (DoS) or Distributed Denial of Service (DDoS): In DoS a server is bombarded with TCP and UDP packets, for accomplishing the task several machines are used to attack the target from multiple locations therefore the DDos is most complex and powerful attack on the internet [1, 2]. Protocol attacks volume-based attacks and application-layer attacks are the three classifications of DDoS attacks that incorporate TCP SYN, ICMP flood and UDP flood as common attacks [4].

Miscellaneous Attacks: Various attacks are present that are used to capture keyboard key press sequences and gets activated when found the keywords Paypal, Yahoo etc. Spam emails are conducted as legitimate email which gathers user information through entering details like bank & personal information, debit-credit card details. All these information is employed as chargeable on the dark web.

II. BACKGROUND

Machine Learning Based: Apart from Naïve Bayesian Classifier **e**very machine learning algorithms imparts with a remarkable performance in detection of botnet traffic. The random forest classifier delivers highest accuracy among all the classifiers. [5] Explored the rest MLA's for the network traffic and found these algorithms performing prevailed for the non malicious traffic. The first ten packets are capable to recognize botnet as contradictory for the entire flow. The best performance of 95% was achieved by monitoring the traffic flow for only 60 seconds.

Deep Learning Based: Deep learning has continuously evolved several solutions in the identification of botnet attacks and rapidly growing with real time solutions. With the network snapshots the traffic behaviour is confined from malware attack on IOT devices and the effect of Mirai and BASHLITE is studied [6] proposed a deep learning based model that can deliver lower false positive rate. Encoding phase has a model, an autoencoder that aimed in the reduction of generous network behaviour. The authors in [7] demonstrated a method for botnet detection based on graph features that focus on the communication structure of node. The model follows LSTM long short term memory for time series evaluation of parameters.

The majority of the dataset offered for botnet detection experience the traffic from both simulated environment and fake constructed traffic which does not produce any kind of real time traffic. Only CTU-13 dataset meets the terms and was detained in the year 2011 as a traffic dataset. CTU -13 is comprised of 13 scenarios and consists of combination of traffic with different malware mockups.

The dataset scenario description is shown below in TableI.

Table I: Depiction of dataset scenario

Id	IRC	SPAM	CF	PS	DDoS	FF	P2P	US	HTTP	Note
1	√	√	√							
2	√	√	√							
3	√			√				√		
4	√				√			√		UDP and ICMP DDoS.
5		√		√					√	Scan web proxies.
6				√						Proprietary C&C. RDP.
7									√	Chinese hosts.
8				√						Proprietary C&C. Net-BIOS, STUN.
9	√	√	√	√						
10	√				√			√		UDP DDoS.
11	√				√			√		ICMP DDoS.
12							√			Synchronization.
13		√		√					√	Captcha. Web mail.

The dataset scenario is widely imbalanced as shown in table II. The scenarios refer to the different traffic distribution

Table II: Imbalance in the dataset

Scenario	Background Flows (%)	Botnet Flows (%)	Normal Flows (%)	Total Flows
1	97.47	1.41	1.07	2,824,636
2	98.33	1.15	0.5	1,808,122
3	96.94	0.561	2.48	4,710,638
4	97.58	0.154	2.25	1,121,076
5	95.7	1.68	3.6	129,832
6	97.83	0.82	1.34	558,919
7	98.47	1.5	1.47	114,077
8	97.32	2.57	2.46	2,954,230
9	91.7	6.68	1.57	2,753,884
10	90.67	8.112	1.2	1,309,791
11	89.85	7.602	2.53	107,251
12	96.99	0.657	2.34	325,471
13	96.26	2.07	1.65	1,925,149

III. METHODOLOGY

This work illustrates an innovative strategy to differentiate botnets inside a virtualised domain; specifically the sort of virtualized condition that may be found inside a cloud specialist organization. Virtualisation compositional structure ideas are starting to perceive focal points of moving the knowledge of an edge gadget into the cloud. This has prompted the development of home system entryways as a help [9] which moves the control of these gadgets from the client to the cloud supplier. This wok focusing on network based botnet detection and capturing of intrinsic and extrinsic attacks on VM is defined by Network based detection process but to identify how vulnerable and threat full these attacks are cannot be defined by such hence to identify type of botnets, This work defines auxiliary approach to classify vulnerability of BotNet into attack weights and frequency if attack depends upon the data available for the approach. For a thriving real time detection of botnet a test environment based model is essential before employ in real time applications. As the existing dataset is lacking in imitation of a real time traffic. Hence only CTU-13 dataset complies for the same. Pursuing the descriptive analytics and feature column analysis individually on dataset, CTU was loaded. Zero variance and visually extraneous columns were dropped. Absent values in some of the columns were attributes.Strategy1, strategy2, and strategy3 datasets were created at the end. Feature selection methods are applied on the three data sets with three different statistical tests. After execution 15 different experiments and two machine learning models achieved. Random forest and decision tree classifier was run on the obtained features to identify the relevant dataset for further consideration. Following strategy 3 data balancing techniques like undersampling with Random under Sampling and NearMiss variants, oversampling with Random Over Sampling, SMOTE and ADASYN variant, oversampling followed by undersampling by SMOTETomek and SMOTEENN, ensemble learning among 6 bagging classifiers and XGBoost classifier was carried out on the dataset.

Finally, for finding the efficient trained model on validation dataset the metric evaluation was performed on each of the experiments. For the performance evaluation the absolute model is run on test dataset and the model will go through training , validation and testing phases for experiments.

Step1. Load the dataset consists of 2,824,636 observations where 40,961 are created by the bot.

	StartTime	Dur	Proto	SrcAddr	Sport	Dir	DstAddr	Dport	State	sTos	dTos	TotPkts	TotBytes	SrcBytes	Bot
0	2011/08/10 09:46:53.047277	3550.182373	udp	212.50.71.179	39678	<->	147.32.84.229	13363	CON	0.0	0.0	12	875	413	0
1	2011/08/10 09:46:53.049843	0.000883	udp	84.13.246.132	28431	<->	147.32.84.229	13363	CON	0.0	0.0	2	135	75	0
2	2011/08/10 09:46:53.049895	0.000326	tcp	217.163.21.35	80	<?>	147.32.86.184	2063	FA_A	0.0	0.0	2	120	60	0
3	2011/08/10 09:46:53.053771	0.056966	tcp	83.3.77.74	32882	<?>	147.32.85.5	21657	FA_FA	0.0	0.0	3	180	120	0
4	2011/08/10 09:46:53.053937	3427.768066	udp	74.89.223.204	21278	<->	147.32.84.229	13363	CON	0.0	0.0	42	2856	1596	0

Fig. 1. Data Set table from notebook

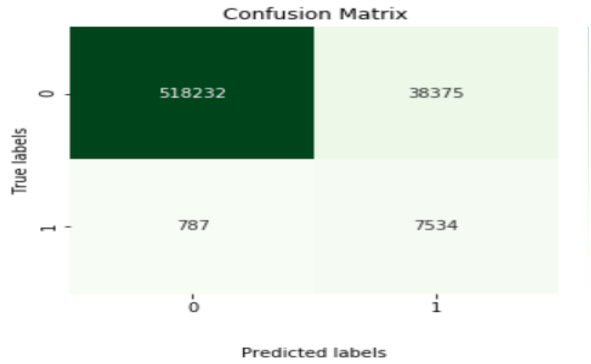
Step2. Data Cleaning including encoding, standardizing the variables

	Dur	TotPkts	SrcBytes	Bot	State_1	State_2	State_3	State_4	State_5	State_6	Proto_1	Proto_2	Proto_3	Dir_1	Dir_2
0	3550.182373	12	413	0	0	0	0	0	0	0	0	0	1	1	0
1	0.000883	2	75	0	0	0	0	0	0	0	0	0	1	1	0
2	0.000326	2	60	0	0	0	1	0	0	0	0	1	0	0	1
3	0.056966	3	120	0	0	0	1	0	0	0	0	1	0	0	1
4	3427.768066	42	1596	0	0	0	0	0	0	0	0	0	1	1	0

Fig. 2. Filtered table of dataset

The data is extremely unbalanced (there are more than 1 million instances of normal traffic and about 35K of bot-created traffic). Balancing of the classes by oversampling the minority classes apply imblearn library.

Step 3. Run Proposed Model of M-XG Boost and Evaluate classification accuracy



The model gives us about 93% precision. However, exactness isn't the entire picture: While 90.54% of the bot-produced traffic was effectively distinguished all things considered, 6.89% of the ordinary traffic was wrongly recognized as bots. For a few, that is not really great, yet it despite everything gives us a ton.

IV. EVALUATION METRICS

The performance and accuracy of the model is evaluated by the classification report and roc curve as a parameter. The classification report is a method from slearn metrics that gives accuracy, precision, recall, and f1-score. Simultaneously, it also demonstrates the confusion matrix. ROC stands for receiver operating characteristic and is useful in acquiring the visualization of trade off during the model training. It represents true and false positive rates at various thresholds. For the baseline setup models RFC and DT classifier are considered with three different configurations of the dataset .The first setup included all the columns obtained after performing column dropping, one-hot encoding, and scaling. The variance thresholding is executed in the second step through dropping of columns that appeared with extreme low variance in the first setup. The third setup consists of analysis of class labels through the frequency distribution at each column with a decrease in number of class. Based on the result it can be stated that random forest performs better than decision tree across all strategies.The trained dataset achieves the size background: 97.47%, botnet: 1.45%, normal: 1.08%. It can be stated that with less features the more prediction could be achive or remained the same.

V. RESULT ANALYSIS

The consequences of grouping utilizing (modified) M-XGBoost classifier and contrasted it and four other machine learning calculations, including Irregular Woodland, SVM, GBDT. The principle examination measures are characterization exactness (Eq. (1)), bogus positive rate (Eq. (2)) and calculation running time.

$$Accuracy = \frac{FP+TN}{TP+FP+TN+FN} \quad (1)$$

$$FPR = \frac{FP}{FP+TN} \quad (2)$$

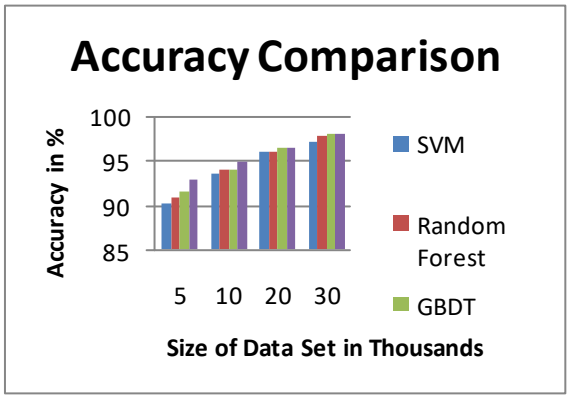
Table –IIIConfusion Matrix

	Predicted Bot Attack	Normal Prediction	Total count
Original Bot	True Positive	False Negative	Positive
Original Normal	False Positive	True Negative	Negative
Sum	(Positive)'	(Negative)'	P+N(P'+N')

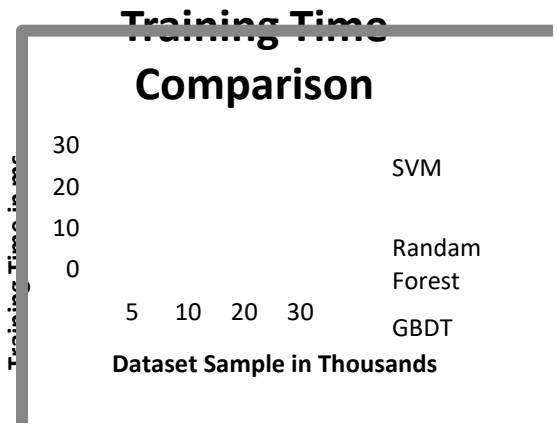
As appeared in graph1, our proposed classifier can accomplish the most elevated precision on account of various informational collections. Refer table IV for subtleties, the outcome in Table IV shows that the exactness of our proposed classifier is about 98%. In any case, the exactnesses of Random Forest, GBDT and SVM are about 94.85%, 96.33%, and 92.88%, which are lower than M- XGBoost classifier individually refer graph 1 . Subsequently, we can see that our proposed classifier accomplishes the best exhibition in precision. As appeared in graph our proposed classifier can accomplish the least bogus positive rate on account of various information sets. See table IV for subtleties, in the bogus positive rate, the calculation of M- XGBoost is 0.009, which is not exactly the FPR of GBDT (FNR is 0.012), SVM (FNR is 0.021), and random forest (FNR is 0.018). For runtime, as appeared in Table IV, the running season of M- XGBoost is 10.02ms, which is just higher than the random forest is 17.25ms, while different calculations (the running season of GBDT is 15.66ms, SVM is 20.22mss) have higher running time than M- XGBoost refer graph 2 . Be that as it may, the presentation of arbitrary woods is more regrettable than M-XGBoost as far as bogus positive and exactness. Along these lines, in general, our model has the best far reaching execution in three viewpoints.

Table-IV Algorithm Comparison

Algorithm	FP Rate	Training Time	Accuracy in %
SVM	0.021	20.22 ms	92.88 %
Random Forest	0.018	17.25 ms	94.85 %
GBDT	0.012	15.66 ms	96.33 %
M-XGBoost	0.009	10.02 ms	97.99 %



Graph 1. Accuracy Comparison with data set applied



Graph 2. Training Time Comparison with data set applied

VI. CONCLUSION

Botnet are assaulting the web terrifically, due to the bot attacks the network services and resources are proliferated. Hence a real time solution is required for the detection of botnet attacks. This work utilizes the machine learning approach for the detection of botnet in network traffic. After reviewing the existing methods it was found that only CTU dataset is best suitable for real time traffic analysis. The data imbalance problem is found in the dataset and to solve these issues under sampling and oversampling variants are considered for experiments and found that the oversampled results in a low performance. While a better prediction with more accuracy was achieved by under sampling. The ensemble learners were trained on subset of data for an auxiliary development through bagging and boosting. An equal performance is attained for balanced bagging classifier and balanced random forest classifier for the botnet class detection. By looking at execution, we can see that the new methodology of M-XGBoost calculation has higher exactness and lower bogus positive rate than different calculations. Moreover, M-XGBoost distinguishes exceptionally rapidly and can adjust to the fast Internet condition. In particular, M-XGBoost have versatility and is appropriate to the cloud with nonstop growing system scale.

REFERENCES

1. M. Feily, A. Shahrestani and S. Ramadass , "A survey of botnet and botnet detection", in *SECURWARE '09 Proc. 2009 3rd Int. Conf. Emerging Security Information, Systems and Tech.*, June 18 – 23, 2009, pp. 268 – 273.
2. P. Amini, M. A. Araghizadeh, R. Azmi, (2016) "A survey on botnet: classification, detection and defense", *in 2015 Int. Electronics Symp. (IES),* Sep. 29 – 30, 2016, DOI: 10.1109/ELECSYM.2015.7380847.
3. M. Mahmoud, M. Nir and A. Matrawy,(2015) "A survey on botnet architectures, detection and defenses", in Int. Journal of Netw. Security, vol. 17, no. 3, pp. 272– 289, May 2015
4. C. Douligeris and A. Mitrokotsa(2004), "DDoS attacks and defense mechanisms: A classification", Jan. 2004, DOI: 10.1109/ISSPIT.2003.1341092
5. Ramachandran, N. Feamster and D. Dagon,(2006) "Revealing botnet membership using DNSBL counter-intelligence", in *Proc. of the 2nd USENIX: Steps to Reducing Unwanted Traffic on the Internet*, San Jose, CA, USA, July 7, 2006, pp. 49–54
6. Y. Meidan (2018)., "N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders", in *IEEE Pervasive Comput. 2018*, Jul. – Sep. 2018, vol. 17, pp. 12 – 22, DOI: 10.1109/MPRV.2018.03367731
7. M. Stevanovic and J. Pedersen,(2014) "An efficient flow-based botnet detection using supervised machine learning", in *2014 Int. Conf. Comput., Netw. and Comm. (ICNC)*, 3-6 Feb. 2014, DOI: 10.1109/ICCNC.2014.6785439
8. S. Garcia(2014), "An empirical comparison of botnet detection", in *Computers and Security*, vol. 45, pp. 100-123, Sep. 2014, doi: 10.1016/j.cose.2014.05.011
9. Dillon,and T. Winters(2014), "Virtualisation of Home Network Gateways" Computer, vol. 47, number 11, IEEE, pp.62-65, November 2014

AUTHORS PROFILE

Yogita Barse, pursuing M.E. in Computer Science Engineering from Indore Institute of Science & Technology, Indore (M.P.) ,I have Completed my Bachelor of Engineering in Computer Science from Shri Dadaji Institute of Science & Technology, Khandwa (M.P.), India. My research interests are Network Security and Artificial Intelligence..

Deepak Agrawal, I am Deepak Agrawal working as an Assistant Professor in department of Computer Science & Engineering, Indore Institute of Science & Technology, Indore (M.P.) , I have published more than 20 papers in International , Scopus Journal and author of two technical books.