International Conference - Innovation-2021-Innovation-2021
International Journal of Scientific Research in Computer Science, Engineering and
Information Technology ISSN : 2456-3307 (www.ijsrcseit.com)
Volume 8, Issue 3, May-June-2021

Machine Learning Based Botnet Detection

Shubham Gour[1], Yogesh Bhosle[1], Onkar Jagtap[1], Pratik Nirmale[1], Prof. Monika Dangore[2]

[1]Department Computer Engineering, Dr. D. Y. Patil School of Engineering, Lohegaon, Pune, Maharashtra, India

[2]Professor, Department Computer Engineering, Dr. D. Y. Patil School of Engineering, Lohegaon, Pune, Maharashtra, India

ABSTRACT

Botnet term was coined when multiple networks of bots came into existence. It is a number of Internet-connected devices, which run single or multiple bots. Botnets can be used to perform Distributed Denial-of-Service attacks, sending spams, and allowing attackers to gain unauthorised access on connections Command and control software is used by the Owner (BotMaster) to control the botnet. This paper discusses the accuracy of the prediction of Botnet detection using different models.

Keywords - Botnet, XGBoost, NaiveBayes, DDOS, Decision Tree, Random Forest, Network Traffic.

I. INTRODUCTION

A bot is an automated program which runs over the internet, some run automatically, while some run when they are triggered by specific input. Internet connected devices are infected with a piece of software that is bot. These internet connected devices are nothing but the botnet. After infection, these internet connected devices steer the instruction commanded by the owner of Botnet known as Bot Master/Bot Herder in 4 phases.

Following are the phases of the botnet infection:

Phase 1 Infection Initialization

A- “Social media” posts targeted by cybercriminal, In the first instance cybercriminal will post a malicious link on social media websites like hoax advertisements, shammed icons etc. When users perform any action on these websites, their action proved to be erroneous, as the current page is redirected to a malicious website, where the software gets installed which was already planted by the BotMaster.

B- “Infection method” approach is followed by the cybercriminals. In this “Email Phishing” tactics are being used to lure users on malicious websites as the user gets redirected when a link is being clicked, and their system gets compromised.

C- “Email Attachments” cybercriminals embody malicious pieces of software with an email, which gets downloaded once clicked and infects the whole system.

Phase 2 Connection to C2C Server

System manifests a connection with a command-and-control (C & C) server which establishes unauthorised connection periodically or may consummate upon infecting the system with malicious activity. Any infected machine liaising with a C & C server will comply to launch a coordinated attack. e.g P2P, TELNET, IRC

Phase 3 Control

Cybercriminal (BotMaster) superintends the command and control of botnets for remote process execution by installing botnets on compromised machines. BotMasters uses Tor/shells to hide their tracks by hiding their identities via proxies to disguise their IP addresses.

Phase 4 Multiplication

Attacks in the first 3 phases are incessanted by Botmasters to infect copious internet devices by malicious use of botnet by promulgating fraud, spam emails, DDOS, keylogger, Miria botnet etc. Most recent attack was the “ Kashmir Black”, an active botnet comprehending thousands of compromised systems across 30 countries and exploiting dozens of vulnerabilities by targeting their CMS. It is believed that the campaign of the “Kashmir Black” started around the end of November 2019 and was trained to target CMS platforms like Vbulletin, Opencart, Yeager, Joomla!, WordPress. Thus after knowing these vicious internet attacks which happen on a daily basis. We decided to counter this issue by implementing an ML model. In this paper we are going to fill up the canvass of loopholes and vulnerabilities with our ML model. To grasp the enormous nature of Machine Learning models let us first know about the basic model of Botnet. Figure 1 stages a basic model of botnet in which botmaster is directly or indirectly connected to every other entity such as server, bots, benign hosts through two way communication.

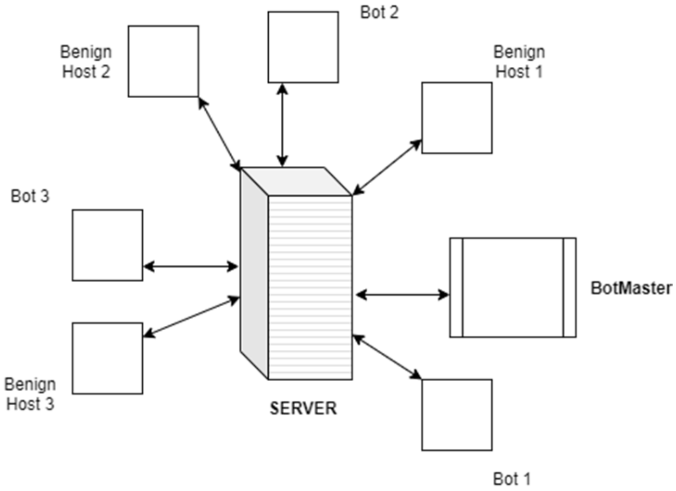


Fig 1.Model of Botnet

II. METHODS AND APPROACHES

In order to detect a botnet we must apply the correct method and follow feasible approaches.

A. XGBoost

Boosting is a sequential technique which follows the principle stated by the ensemble model. It has a set of weak learners which helps to ameliorate prediction accuracy. At any instant s model outcomes are weighed on previous instant t-1. outcomes which get predicted correctly assigned as first know about the basic model of Botnet. Figure 1 stages a basic model of botnet in which botmaster is directly or indirectly connected to every other entity such as server, bots, benign hosts through two way communication.

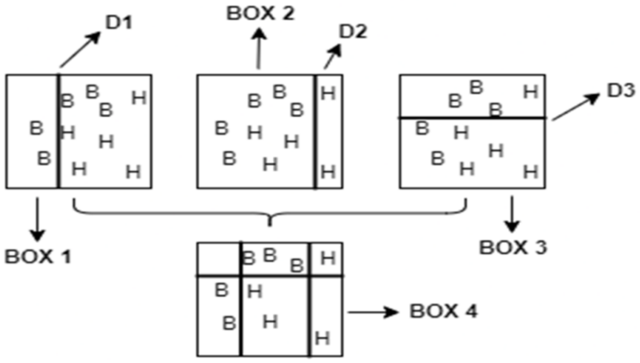


Fig 2.XGBoost

A lower weight and which got miss-classified weighted higher.

Four classifiers (in 4 boxes), shown above, are trying to classify bots B and Benign Host H classes as homogeneously as possible.

1. Box 1- First classifier (a decision stump) makes a vertical line (split) D1. It says anything to the left of D1 is B and anything to the right of D1 is H. However, this classifier misclassified three B points. Decision Stump is a Decision Tree model which only splits off at one level, so the final prediction is based on only one feature.
2. Box 2: The second classifier gives more weight to the three B misclassified points (see the bigger size of B) and creates a vertical line at D2. Again it says, anything to the right of D2 is H and left is B. Still, it makes mistakes by incorrectly classifying three H points.
3. Box 3: Again, the third classifier gives more weight to the three H misclassified points and creates a horizontal line at D3. Still, the classifier fails to classify points correctly.
4. Box 4: Weak weighted classifiers combination of (Box 1, Box 2 and Box 3). It does a good job by classifying all points correctly. This is the basic idea how this algorithm will help us to identify botnets.

B. Naive Bayes Algorithm

The Naive Bayes classifier is a probabilistic machine learning model that's used for classification task. The core of the classifier is based on the Bayes theorem

$$P\,(A|B) = (P\,(B|A)\,P\,(A))/P\,(B)$$

The Naive Bayes classifier is a probabilistic machine learning model that's used for classification tasks. The core of the classifier is based on the Bayes theorem.

It is mainly used in sentiment analysis, filtering the spam, etc. Naive Bayes is fast and easy to implement but the drawback of this is that the requirement is that the predictors need to be independent. In most of the real life cases, the predictors are dependent; this hinders the performance of the classifier.

C. Decision Tree Algorithm

In machine learning Classification is a two-step process that is learning and prediction. The model is developed based on given training data in learning steps. The model is used to predict the response for given data in the prediction step. Decision Tree Algorithm (DTA) comes under Supervised Learning Algorithm (SLA). DTA can be used for solving regression and classification problems unlike other SLA. The goal of using a Decision Tree is to create a training model that can be used to predict the class or value of the target variable by learning simple decision rules inferred from prior data (training data). In Decision Trees, we start from the root for record from the prediction of a class label. We compare the values of the root attribute with the record's attribute. On the basis of comparison, we follow the branch corresponding to that value and jump to the next node.

D. Random Forest

Random Forest is a popular machine learning algorithm which belongs to the technique of supervised learning. Random Forest Algorithm can be used for both Classification and Regression problems in Machine Learning. Random forest is based on ENSEMBLE LEARNING, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. In simpler terms, Random Forest uses Decision trees in a randomized way. Random forest requires less training time as compared to other algorithms and high accuracy output can be produced using Random Forest. Even for the large dataset random forest runs efficiently. Implementation Steps for random forest are as

1. Data Pre-processing
2. Fitting R.F algorithm to the Training set
3. Predicting the test result
4. Test accuracy of the result
5. Visualizing the test set result.

ML Approach

Machine learning has various applications and methods to solve real world problems in discrete domains. This is possible due to abundant data spread across the network, significant furtherance of ML techniques, and advancement in computing capabilities. In the figure we discussed the components which are used to build a robust ML model for a given networking model.ML has been applied to dispense its flexible nature to solve real world complex problems in network operations and other sectors

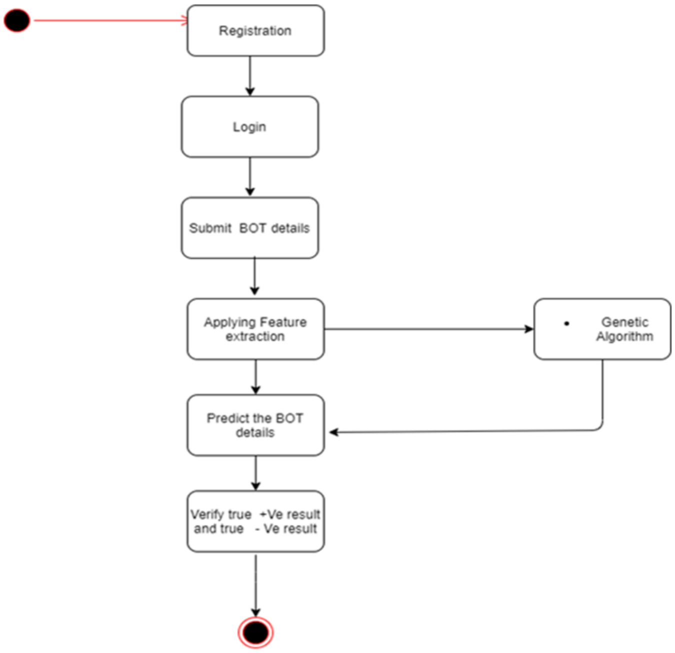


Fig 3. ML based solution.

In survey we found out that perplexed problems across different network technologies can be unraveled by using diverse ML techniques which is an injunction with diverse application of Machine Learning.There are fragments like QOE , QOS management, traffic prediction, congestion control, routing and classification management of networking which we have discussed in our paper to get the insights, scientific challenges and extent of ML in networking. Every effort is accountable and holds the responsibility to push the barriers of automatic network operations and their activities by using the features of ML in networking.

III. RESULTS AND DISCUSSION

We propose a machine learning based botnet detection system that is shown to be effective in identifying P2P botnets. Our approach extracts convolutional versions of effective flow-based features, and trains a classification model by using a feed-forward artificial neural network. The experimental results show that the accuracy of detection using the convolutional features is better than the ones using the traditional features. It can achieve 94.7% of detection accuracy and 2.2% of false positive rate on the known P2P botnet datasets. Furthermore, our system provides additional confidence testing for enhancing performance of botnet detection. It further classifies the network traffic of insufficient confidence in the neural network. The experiment shows that this stage can increase the detection accuracy up to 98.6% and decrease the false positive rate up to 0.5%.

STEP 1

Admin has to enter their login credentials in order to detect botnets using various models . This type of phase plays a crucial role to avoid any vulnerabilities of the system and also to strengthen the security and integrity of the system.

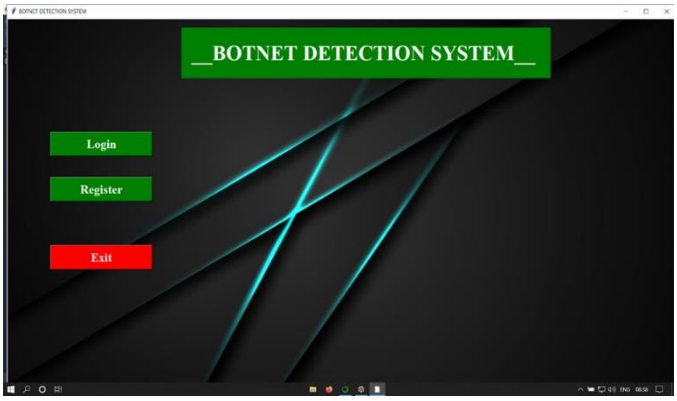


Fig 4.Registration Page

Once you register yourself in the Botnet detection system you are ready to Login with the information you provided in the Registration page .

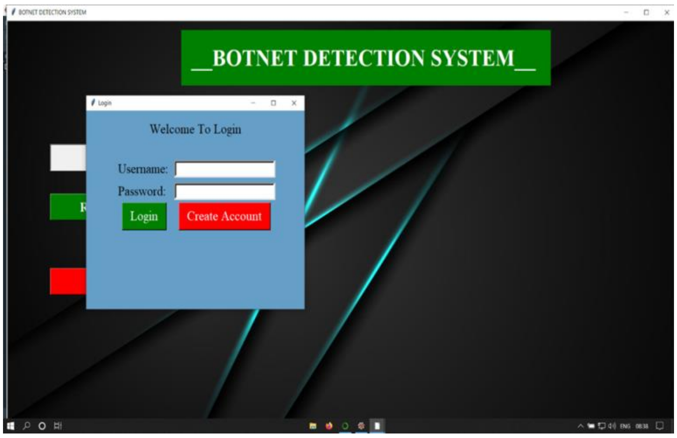


Fig 5.Login Page

STEP 2

After being authenticated by the system you will be allowed to view the different models, which will help us to choose the better and feasible model to detect botnets.

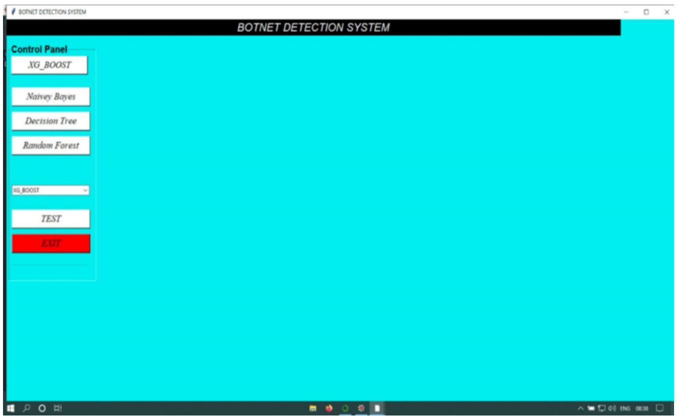
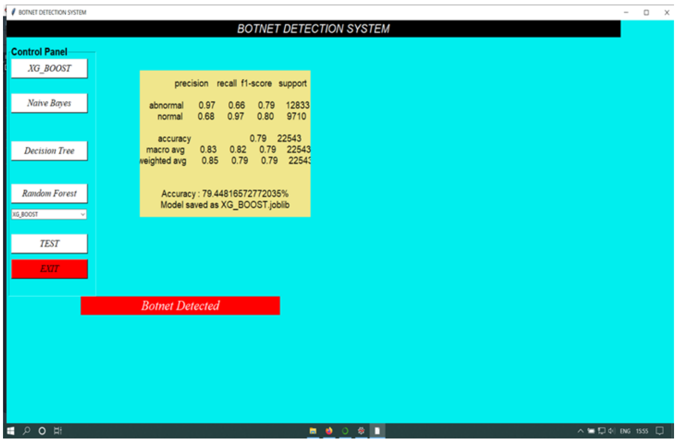


Fig 6.Algorithms

STEP 3

First we choose the Algorithms to train our model through the training phase which uses KDD_Cup Dataset. After selecting the algorithm and training the model we move towards the drop down list provided below in the Botnet detection page. We select the algorithm from the drop down list. To detect a Botnet, System must be assigned with a dataset which has plenty of fields with information full of network logs and traffic.



```
              precision    recall  f1-score   support

    abnormal       0.97      0.66      0.79     12833
      normal       0.68      0.97      0.80      9710

    accuracy                           0.79     22543
   macro avg       0.83      0.82      0.79     22543
weighted avg       0.85      0.79      0.79     22543


Accuracy : 79.44816572772035%
Model saved as XG_BOOST.joblib
```

Botnet Detected

Fig 7.Result

Comparative analysis of botnets with different ML techniques gives us the idea that using a single model for detecting botnets is of no use as the technology will keep on growing and bots will become smarter. So to rely on a single model is not a smart move. We implemented different models so that we can demonstrate the different prediction accuracy achievable by using different algorithms.

XGBoost- 79.448%

Naive Bayes- 45.029%

Decision Tree-79.532%

Random Forest-76.227%

IV. FUTURE SCOPE

We intend to implement this model on a larger scale by improving its detection to work in real time. The proposed model detects the botnet but is not capable of handling large datasets. Real time interpretation of n/w logs and monitoring the accuracy in real time is yet to be achieved. As the technology improves and more and more new tools will be available to handle loads of data with high accuracy of predicting botnet detection.

V. CONCLUSION

This paper examines various techniques and methods to deal with botnets under different situations over different networks and compares different algorithms and their accuracies. The main threat in bot detection is to avoid any loopholes or vulnerabilities in our own system while tracking them to terminate the bot's network before their vicious goal is achieved by their botmaster.We have successfully implemented the different models and achieved higher accuracy in prediction of existence of Botnet in a system. 80% of data is used to train the model and the remaining 20% of data is used to test the model

VI. REFERENCES

[1]. Sudipta Chowdhury1*, Mojtaba Khanzadeh1, Ravi Akula1, Fangyan Zhang2, Song Zhang2, Hugh Medal1, Mohammad Marufuzzaman1, Linkan Bian1” Botnet detection using graph-based feature clustering”.

[2]. Zhuang and J. M. Chang, “PeerHunter: Detecting peer-to-peer botnets through community behavior analysis”.

[3]. S. Lagraa, J. François, A. Lahmadi, M. Miner, C. Hammerschmidt and R. State, "BotGM: Unsupervised graph mining to detect botnets in traffic flows," 2017 1st Cyber Security in Networking Conference (CSNet), Rio de Janeiro, 2017, pp. 1-8, doi: 10.1109/CSNET.2017.8241990.

[4]. Sara Khanchi, Ali Vahdat, Malcolm I. Heywood, A. Nur Zincir-Heywood,"On botnet detection with genetic programming under streaming data label budgets and class imbalance", Swarm and Evolutionary Computation, Volume 39, 2018, ISSN 2210-6502

[5]. Jeeyung Kim, Alex Sim, Jinoh Kim, Kesheng Wu,” Botnet Detection Using Recurrent Variational Autoencoder”.

[6]. Hagan, M., Kang, B., McLaughlin, K., & Sezer, S, “Peer Based Tracking using Multi-Tuple Indexing for Network Traffic”.

[7]. Raouf Boutaba 1, Mohammad A. Salahuddin 1, Noura Limam 1, Sara Ayoubi 1, Nashid Shahriar 1, Felipe Estrada-Solano1,2 and Oscar M. Caicedo 2 “Survey on machine learning for networking: evolution, applications and research opportunities”.

[8]. E. M. Hutchins, M. J. Cloppert, and R. M. Amin, “Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains,” Inf. Warfare Security Res., vol. 1, no. 1, p. 80,2011.

[9]. S. Chen, Y. Chen and W. Tzeng, "Effective Botnet Detection through Neural Networks on Convolutional Features," 2018 17th IEEE International Conference on Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), New York, NY, 2018, pp. 372-378, doi: 10.1109/TrustCom/BigDataSE.2018.00062.

[10]. B. Alothman and P. Rattadilok, "Towards using transfer learning for Botnet Detection," 2017 12th International Conference for Internet Technology and Secured Transactions (ICITST), Cambridge, 2017, pp. 281-282, doi: 10.23919/ICITST.2017.8356400.

[11]. G. Vormayr, T. Zseby and J. Fabini, "Botnet Communication Patterns," in IEEE Communications Surveys & Tutorials, vol. 19, no. 4, pp. 2768-2796, Fourthquarter 2017, doi: 10.1109/COMST.2017.2749442.

[12]. H. Dhayal and J. Kumar, "Botnet and P2P Botnet Detection Strategies: A Review," 2018

International Conference on Communication and Signal Processing (ICCSP), Chennai, 2018, pp. 1077-1082, doi: 10.1109/ICCSP.2018.8524529.

[13]. C. Czosseck, G. Klein and F. Leder, "On the arms race around botnets - Setting up and taking down botnets," 2011 3rd International Conference on Cyber Conflict, Tallinn, 2011, pp. 1-14.

[14]. K. Alieyan, M. Anbar, A. Almomani, R. Abdullah and M. Alauthman, "Botnets Detecting Attack Based on DNS Features," 2018 International Arab Conference on Information Technology (ACIT), Werdanye, Lebanon, 2018, pp. 1-4, doi: 10.1109/ACIT.2018.8672582.

[15]. W. Zhang, Y. -J. Wang and X. -L. Wang, "A Survey of Defense against P2P Botnets," 2014 IEEE 12th International Conference on Dependable, Autonomic and Secure Computing, Dalian, 2014, pp. 97-102, doi: 10.1109/DASC.2014.26.

[16]. W. Sun and H. Gou, "The Botnet Defense and Control," 2011 International Conference of Information Technology, Computer Engineering and Management Sciences, Nanjing, Jiangsu, 2011, pp. 339-342, doi: 10.1109/ICM.2011.218.