An IoT-based botnet defence honeypot with machine learning-based detection framework

[1]E Krishna,Assistant Professor,
[2]Kalvala Sadanandam ,Associate Professor,
[3]M Naveen Kumar,Assistant Professor,
Department of CSE Engineering,
Pallavi Engineering College,
Kuntloor(V),Hayathnagar(M),Hyderabad,R.R.Dist.-501505.

***Abstract*— With the huge increase of IoT botnet DDoS assaults in recent years, IoT security has now become one of the most worried subjects in the world of network security. A number of security measures have been offered in the field, however they still lack in terms of dealing with newly developing varieties of IoT malware, known as Zero-Day Attacks. In this research, we describe a honeypot-based strategy which leverages machine learning techniques for malware detection. The IoT honeypot produced data is utilized as a dataset for the effective and dynamic training of a machine learning model. The technique might be viewed as a fruitful beginning towards battling Zero-Day DDoS Attacks which now has arisen as an open difficulty in safeguarding IoT from DDoS Attacks.**

***Keywords*— Zero-Day DDoS Attack; Machine Learning; IoT Honeypots; IoT Botnets**

INTRODUCTION

There's been an increase in DDoS attacks due to IoT, a network of networked devices without human involvement. [1] The security of Internet of Things (IoT) devices is more vulnerable than that of traditional desktop PCs. Because of this, IoT-based botnet assaults are becoming more common [7]. An IoT network has been infected with malware, resulting in the creation of a botnet, which is a collection of hacked IoT devices [2]. According to a recent study, there are more than 6 billion IoT devices on the earth, which means that fraudsters will not be able to escape undetected. Hundreds of thousands of pieces of malware have been discovered throughout the years, with the majority appearing in 2017 [5].

When a honeypot is used to lure in attackers for the purpose of gathering information about the attacking agent like malware for a DDoS assault, that's exactly what it is: a trap. By imitating a weakness that may be exploited by an attacker, this device can be used to compromise the main server. When it monitors the activity of an attacker and itself, it is able to gather information such as IP addresses, MAC addresses, ports, types of devices targeted, malware executables, and their instructions [27]. Honeypots have been shown to be a valuable tool in the fight against malware and its variations in recent years in the realm of computer security. The 'Deception Toolkit,' created by Fred Cohen in 1998 [28], originally appeared in the late 1990s and was made accessible to the general public and for commercial use in order to combat worms, which are self-replicating programmes.

Honeypots come in a variety of shapes and sizes, making them suitable for a wide range of uses. Depending on how much contact it permits with the attacker, it may be characterised as one of many types. This depends on the quantity of data that has to be gathered. As a result, it is divided into low- and high-interaction honeypots. Honeypots may also be categorised based on the goal they are trying to achieve, such as doing research to learn about potential threats and flaws in the system, or safeguarding the company's assets in real time to enhance overall security, known as Production Honeypots. Because they don't compromise IoT devices, honeypots are a good defence against Zero-Day DDoS Attacks [29].

Traditional honeypots and IoT honeypots are two distinct types of honeypots. As a result of the variety of IoT devices, traditional honeypot designs are homogeneous (mostly x86 and x86-84) whereas IoT honeypot architectures are diverse (mainly ARM).

Using a honeypot architecture, we have been able to capture a number of attempts to implant malware on the IoT device. We may utilise log files as input to the machine learning model we're employing for training purposes by analysing the data. It is possible to train the model by employing both known and undiscovered malware types by using honeypots instead of using a restricted number of datasets [13].

IoT device security risks are detected and predicted utilising relevant machine learning algorithms and methods in our solution. Unsupervised and supervised learning algorithms are the two most common types of algorithms. The assignment of classification labels during the training phase is required for supervised learning in order to predict the labels if the related characteristics are roughly the same. On the other hand, in unsupervised learning [6], labels aren't necessary; instead, classification is based on the similarity of the dataset's characteristics. Because an expert is required to develop the rules and assign the labels, we opted for an unsupervised learning algorithm in our approach to avoid involving humans in the process. These include clustering, anomaly detection, and neural networks, among others. Both a classification issue and a clusterization problem may be used to describe malware detection. Classification is an example of a problem that can be solved using supervised learning since it includes known examples of data. Unknown malware kinds are clustered into a number of clusters using an unsupervised learning technique in the clusterization problem [8].

Moreover, the advantage of using machine learning for the detection of malware lies in its ability to generate a lesser number of false positives and false negatives as compared to other anomaly detection methods [4].

RELATED WORK

There are a number of honeypot-based defences for DDoS in the literature. Some of these systems made use of the signature matching method as a detection framework [16]. The honeypot uses signatures found in the malware's produced log files to detect it [18]. In order to cope with an unknown and greater variety of malware families, this detection method could only deal with stored signatures and their changes. Alternatively, there is anomaly-based detection [12], which doesn't employ rules, but rather a threshold for regular user behaviour is defined and any divergence from it results in a statement of probable hostile activity. Such systems have a high risk of false positives due to the fact that attackers may mimic legitimate activity as well. This challenge can be solved by a machine learning-based solution since it can learn and teach itself over time. By using current and precise data to train the model, a more accurate categorization may be accomplished with fewer false positives. The honeypot's dynamic data may be better used with the help of machine learning in order to better forecast future assaults.

Many supervised learning algorithms, such as SVM and NaveBayes, have been suggested to detect DDoS attacks using machine learning approaches based on statistical feature selection [15,17]. To choose the right characteristics from the dataset, these algorithms need substantial network experience and are often restricted to just one or a few DDoS routes. In addition, they need to maintain the system updated on a regular basis so that it can handle a variety of conditions.

Deep learning models including Convolutional Neural Network (CNN) [22], Recurrent Neural Network (RNN) [25], Long Short-Term Memory Neural Network (LSTM) [23], and Gated Recurrent Unit Neural Network (GRU) [24] have been suggested to identify DDoS attacks using machine learning. It was suggested that a network-based anomaly detection approach gathers network activity snapshots and use deep autoencoders to identify unusual network traffic originating from exploited IoT devices The quantity of data needed to train a deep learning model to get reliable results is enormous. Despite this, their training procedures are exceedingly costly and complicated, and they frequently take a long time to master. Because of their limited resources and inability to provide real-time services to users, IoT devices cannot afford such arduous operations. Even more importantly, new ways of detecting attacks launched from hacked IoT devices and distinguishing between assaults lasting an hour and milliseconds need to be developed.

METHODOLOGY

For the Zero-Day DDoS assaults, we are not only interested in detecting the malware, but also identifying the unknown malware families that are involved. DDoS defences against Zero-Day assaults can't protect against all conceivable varieties of malware infestations since they haven't yet been discovered. A honeypot technique with a detection system based on machine learning solves this problem. In order to catch malware qualities and its method of compromising the security of IoT devices, a honeypot is employed to purposefully draw in attackers and record all the information about it in log files [16]. It is also used to predict abnormal activity based on the log files generated by the honeypot by using a machine learning-based detection framework, which uses a light weighted classification algorithm, preferably an unsupervised one, to classify the training tuples into a malicious one or a normal one.

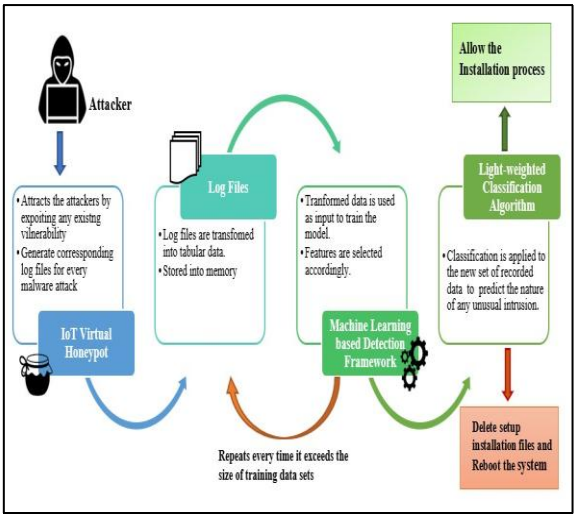
Attacker
•Attracts the attackers by expoiting any existng vilnerability
•Generate corresponding log files for every malware attack
IoT Virtual Honeypot
Log Files
•Log files are transfomed into tabular data.
•Stored into memory
•Tranformed data is used as input to train the model.
•Features are selected accordingly.
Machine Learning based Detection Framework
Light-weighted Classification Algorithm
•Classification is applied to the new set of recorded data to predict the nature of any unusual intrusion.
Allow the Installation process
Delete setup installation files and Reboot the system
Repeats every time it exceeds the size of training data sets

Fig. 1.Process flow for the honeypot-based solution with machine learning based detection framework

Our suggested solution has the following architecture: To begin, an attacker has to get access to an IoT device using a variety of ID and Password combinations over an open port (such as telnet port 23 or 2323) and inject malware into the system. In this case, the honeypot comes into play for purposely enabling the adversary to get over its own defence wall. Log files are used to keep track of all communications between the device and the intruder in order to learn more about the infection and the person who is behind it. New malware families, their variations, types of devices they're aimed at, and their C&C servers' IP addresses, ports, and other details may all be gleaned from these log files. As a result, in order to use our log file data as datasets for training our machine learning model, we must first convert it to a tabular format. Using a memory efficient classifier that requires the least amount of training data to predict meaningful information is preferable to avoid making an IoT device burdened by it [20]. Finally, suitable action is taken in light of the categorization outcome. Fig.1 depicts the suggested solution's whole process flow. The training procedure repeats itself if the training data exceeds the allowed amount, making it dynamic and easy to operate on resource-constrained IoT devices.

IMPLEMENTATION ASPECTS

Implementation is an essential aspect of every unique strategy or concept in order to examine the feasibility and evaluate its efficiency over the already existing comparable solutions. As described in the preceding section, our suggested strategy comprises of various following phases. At each phase, we may use the newest methodologies for the underlying notion to keep our solution updated enough to address the current IoT difficulties. Following are the current advances that took place in recent years in the area of IoT honeypots and real-time machine learning detection which are the two most essential phases employed in our strategy for carrying out the required implementation:

A.IoT Virtual Honeypot:

Our very first step in our recommended strategy is to attract the attackers for purposely exploiting the vulnerability existing in IoT devices. For imitating this behaviour, we need a system or device which can precisely behave as an exploitable IoT device and push the attacker to perform his evil action without having the second thought about the authenticity of the vulnerabilities. Such systems are popularly recognised as IoT honeypots. As indicated above in the introduction depending on the amount of engagement, honeypots may be classed as High Interaction Honeypots (HIH), Low Interaction Honeypots (LIH) and Medium Interaction Honeypots (MIH) which is a mix of both. Since it's infeasible to set up a high interaction honeypot (HIH) for resource-constrained IoT devices, it would be better to chose medium interaction honeypot (MIH) over the other two honeypots. That is the reason why it is known as IoT 'Virtual' honeypot since in this scenario we would be constructing it virtually by imitating the IoT platform utilising IoT communication protocols. The attack techniques including network traffic, payload, malware samples, the toolkit used by the attacker, etc. are then may be captured by the honeypot. There is a list of several newly created IoT honeypots for DDoS detection:

IoTPOT [32]: This honeypot also emulates Telnet services of various IoT devices and comprises of a frontend low interaction responder working with a backend high interaction virtual environment called IoTBOX capable of functioning at multiple CPU architectures.

Telnet IoT honeypot [30]: Telnet server is used for constructing the trap for IoT.

HoneyThing [31]: This honeypot emulates a susceptible modem/router (with RomPager embedded web server) and is TR-069 (CPE WAN Management Protocol) specific.

Dionaea [33]: This honeypot utilises MQTT protocol to replicate the IoT behaviour.

ZigBee Honeypot [34]: This honeypot replicates a ZigBee gateway.

Multi-purpose IoT honeypot [35]: This IoT honeypot focuses on Telnet, SSH, HTTP, and CWMP.

ThingPot [29]: This IoT honeypot is capable of emulating a full IoT platform, rather than a single applicationlayer communication protocol (e.g., Telnet, HTTP, etc). (e.g., Telnet, HTTP, etc.).

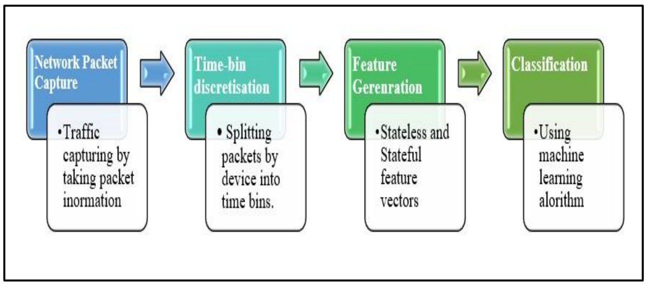
Network Packet Capture
•Traffic capturing by taking packet inormation
Time-bin discretisation
• Splitting packets by device into time bins.
Feature Gerenration
•Stateless and Stateful feature vectors
Classification
•Using machine learning alorithm

Fig. 2.Process flow for the machine learning based detection framework.

The best IoT honeypot is capable of mimicking the whole IoT platform, including all of the supporting application layer protocols, rather than only imitating a few chosen IoT communication protocols. IBM's MQTT (Message Queue Telemetry Transport), XMPP (Extensible Messaging and Presence Protocol), AMQP (Advanced Message Queuing Protocol), CoAP (Constrained Application Protocol), and UPnP (Universal Plug and Play) are some of the most widely used application protocols for Internet of Things (IoT) communication. Representational State Transfer or REST is a frequently used architectural approach in M2M and IoT systems. We can utilise ThingPot, one of the honeypots on this list, to investigate a wide range of malware threats.

B.Detection Framework for Machine Learning in Real Time

Another crucial element in our DDoS detection method is the use of a machine learning-based detection system. Classification may be accomplished using a variety of machine learning methods. The categorization of malware isn't enough for us; we want a machine learning solution that can effectively categorise malware characteristics without creating a lot of false positives in real time. R. Doshi et al., 2018 [17] offered a system for real-time machine learning-based malware detection in IoT devices that has shown to identify malware with an accuracy of 0.99 [17]. IoT botnet assaults have been on the rise in recent years, and this solution is specifically designed to combat them.

Unlike typical computers and smartphones that interact with big web servers, IoT devices communicate with endpoints within a local range. A machine learning approach may be used to monitor IoT traffic for this kind of activity. Data collection is the first phase, followed by feature extraction and, lastly, binary classification, to complete the process. The retrieved features are mostly network behaviours related to the Internet of Things (IoT), including packet length, inter-packet intervals, and protocol, among others. Different attack detection classifiers, including random forests, K-nearest-neighbours, support vector machines, decision trees, and neural networks, are evaluated. It was discovered that random forest, K-nearest neighbours, and neural net classifiers were among the most successful [17]. Many machine learning methods, such as neural networks, may be utilised to improve the accuracy of detecting DDoS in IoT network traffic by using IoT-specific network patterns such as the restricted number of endpoints and the regular time gap between packets.

Traffic capture, grouping, feature extraction, and binary classification are all stages in the process of anomaly detection, which begins with Traffic Capture, finishes with Binary Classification. Recording IP packets from an IoT device, such as one used in a smart home application, and saving the information contained in them is what is meant by traffic capture. Due to the complexity and security hazards involved, gathering DDoS traffic is a difficult undertaking. For the purpose of capturing new malware varieties, it has replicated three of the most prevalent DDoS attack types: TCP SYN flooding, UDP flooding, and HTTP GET flooding.

An IoT device's originating IP address is used to group its packets, which are then subdivided into non-overlapping timestamps.

According on IoT device behaviour, the feature extraction procedure generates stateless and stateful features for each packet. When a packet is transmitted, its flow-independent properties are used to produce stateless features, which are light-weight features that do not need the traffic to be separated by IP source. Stateful features, on the other hand, are concerned with obtaining the aggregated network traffic flow information across short time intervals. Stateless characteristics include packet size and inter-packet interval, whereas stateful features include bandwidth, IP address cardinality, and novelty. A binary classification method such as K-nearest neighbours, random forests or support vector machines is used to separate normal traffic from DDoS traffic flow [36]. There are several steps involved in this procedure, and this diagram illustrates them all. With the extra data that comes from real-world deployments, the use of deep learning classifiers will be more successful.

An IoT honeypot based on the ThingPot [29] may be used to achieve the suggested approach since it is an IoT virtual honeypot capable of capturing multiple botnet binaries by simulating different IoT communication protocols and the complete IoT platform's behaviours. The virtual box should be used to put it on every IoT device in a network in order to maintain it separated from the original IoT platform. Classifiers should be built at the router level rather than on each device owing to IoT restrictions. In addition, the volume of traffic flowing via an IoT device makes it impossible to build a machine learning model on it. Any IoT network simulator may be used to produce enough traffic. To test any IoT-based application, IoT simulators generate an IoT environment and, if necessary, provide storage over the cloud. As a result, if we're utilising the honeypot of our choice, we don't need to worry about IoT simulators. Using bash scripts on Linux, the log files may be converted to the format needed by machine learning models. Machine learning technologies like Microsoft Azure, MATLAB, and others may be utilised in a virtualized environment to complete tasks. This technique may be used to discover real-time machine learning anomalies.

CONCLUSION

In terms of technology, the Internet of Things (IoT) is the most important factor in the modernisation of the actual world. DDoS assaults are also on the rise as a result of the rising amount of cyber attacks. Because of this, Internet Security has shifted its focus to fighting against attacks that leverage IoT as a means of harming network security. IoT botnet assaults have spawned several protection measures, but none have been able to keep up with the ever-evolving threats that they face. In order to identify DDoS attacks in real time, we developed a honeypot-based detection system that makes use of machine learning. ML-based detection frameworks will be able to train their classifiers more efficiently if new malware traits are logged using honeypots. This method should be taken to the next level in order to identify the open difficulties or concerns in real-time situations for the future scope of the project. A cloud server may also be used to handle resource-constrained IoT devices. When we compare our suggested solution to other offered models, we may draw conclusions about its performance.

REFERENCES

1. *K. Chen, S. Zhang, Z. Li,Yi Zhang, Q.Deng, Sandip Ray, YierJin, "Internet-of-Things Security and Vulnerabilities: Taxonomy, Challenges, and Practice" Journal of Hardware and Systems Security, vol. 2, Issue 2, pp. 97–110, (2018).*
2. *W. Zhou, Y. Jia, A. Peng, Y. Zhang and P. Liu, "The Effect of IoT New Features on Security and Privacy: New Threats, Existing Solutions, and Challenges Yet to Be Solved," IEEE Internet of Things Journal. 2018.*
3. *J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications," IEEE Internet of Things Journal, vol. 4, no. 5, pp. 1125-1142 (2017).*
4. *Honeypots and the Internet of Things. Available at https://securelist.com/honeypots-and-the-internet-of-things/78751.*
5. *Hastie, T., Tibshirani, R.,& Friedman, J. Unsupervised learning. In The elements of statistical learning (pp. 485-585). Springer, New York, NY (2009).*
6. *C. Kolias, G. Kambourakis, A. Stavrou and J. Voas, "DDoS in the IoT: Mirai and Other Botnets," in Computer, vol. 50, no. 7, pp. 80-84 (2017).*
7. *Dougherty, J., Kohavi, R., &Sahami, M. Supervised and unsupervised discretization of continuous features. In Machine Learning Proceedings 1995, pp.194-202 (1995).*
8. *Sommer, R., & Paxson, V. (2010, May). Outside the closed world: On using machine learning for network intrusion detection. In Security and Privacy (SP), IEEE Symposium on (pp. 305-316). IEEE (2010).*
9. *M. Anirudh, S. A. Thileeban And D. J. Nallathambi, "Use of Honeypots for Mitigating DoS Attack Targeted on IoT Networks," 2017 International Conference On Computer, Communication And Signal Processing (ICCCSP), Chennai, Pp. 1-4, (2017).*
10. *Rieck, K., Holz, T., Willems, C., Düssel, P., &Laskov, P. (2008, July). Learning and classification of malware behavior. In International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment (pp. 108-125). Springer, Berlin, Heidelberg.*
11. *Bailey, M., Oberheide, J., Andersen, J., Mao, Z. M., Jahanian, F., & Nazario, J. Automated classification and analysis of internet malware. In International Workshop on Recent Advances in Intrusion Detection Springer, Berlin, Heidelberg, pp. 178-197 (2007).*
12. *Binkley, J. R., & Singh, S. An Algorithm for Anomaly-based Botnet Detection. SRUTI, 6, 7-7. (2006).*
13. *Song, Y., Keromytis, A. D., &Stolfo, S. J. U.S. Patent No. 8,844,033. Washington, DC: U.S. Patent and Trademark Office. (2014).*
14. *The New Threat: The IoT DDoS Invasion. https://www.a10networks.com/sites/default/files/resource-files/A10TPS-GR-The_New_Threat_The_IoT_DDoS_Invasion.pdf.*
15. *Zammit, DA machine learning based approach for intrusion prevention using honeypot interaction patterns as training data. University of Malta, 1-55 (2016).*
16. *Pa, Y. M. P., Suzuki, S., Yoshioka, K., Matsumoto, T., Kasama, T., &Rossow, C. IoTPOT: analysing the rise of IoT compromises. EMU, 9, 1(2015).*
17. *Doshi, R., Apthorpe, N., &Feamster, N. Machine Learning DDoS Detection for Consumer Internet of Things Devices, arXiv preprint arXiv:1804.04159 (2018).*
18. *Pa, Y. M. P., Suzuki, S., Yoshioka, K., Matsumoto, T., Kasama, T., &Rossow, C., IoTPot: A novel honeypot for revealing current IoT threats. Journal of Information Processing, 24(3), 522-533 (2016).*
19. *Musca, C., Mirica, E., &Deaconescu, R. Detecting and analyzing zeroday attacks using honeypots. In Control Systems and Computer Science (CSCS), 2013 19th International Conference on (pp. 543-548). IEEE. (2013).*
20. *Hofmann, T. Unsupervised learning by probabilistic latent semantic analysis. Machine learning, 42(1-2), 177-196 (2001).*

21. *Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, pp. 1097–1105 (2012).*
22. *S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol. 9, no. 8, pp. 1735–1780, (1997).*
23. *J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," arXiv preprint arXiv:1412.3555, (2014).*
24. *Yuan, X., Li, C., & Li, X. DeepDefense: Identifying DDoS Attack via Deep Learning. In 2017 IEEE International Conference on Smart Computing (SMARTCOMP) (pp. 1-8). IEEE. (2017).*
25. *Meidan, Y.,Bohadana, M., Mathov, Y., Mirsky, Y., Shabtai, A., Breitenbacher, D., &Elovici, Y. N-BaIoT—Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders. IEEE Pervasive Computing, 17(3), 12-22. (2018).*
26. *Nawrocki, M., Wählisch, M., Schmidt, T. C., Keil, C., &Schönfelder, J. A survey on honeypot software and data analysis. arXiv preprint arXiv:1608.06249. (2016).*
27. *Cohen, F. Special feature: A note on the role of deception in information protection. Computers and Security, 17(6), 483-506. (1998).*
28. *Syversen, J. , U.S. Patent Application No. 11/632,669 (2008).*
29. *Wang, Meng, Javier Santillan, and Fernando Kuipers. "ThingPot: an interactive Internet-of-Things honeypot." arXiv preprint arXiv:1807.04114 (2018).*
30. *Phype. Telnet IoT honeypot. https://github.com/Phype/ telnet-iothoneypot.*
31. *Omererdem. Honeything. https://github.com/omererdem/honeything. [32]Yin Minn Pa Pa, Shogo Suzuki, Katsunari Yoshioka, Tsutomu Matsumoto, Takahiro Kasama, and Christian Rossow. IoTpot: A novel honeypot for revealing current iot threats. Journal of Information Processing, 24(3):522–533, 2016.*
32. *DinoTools. dionaea - catches bugs. https://github.com/DinoTools/ dionaea/blob/master/README.md.*
33. *S. Dowling, M. Schukat, and H. Melvin. A zigbee honeypot to assess iot cyberattack behaviour. In 2017 28th Irish Signals and Systems Conference (ISSC), pages 1–6, June 2017.*
34. *Roy T. Fielding and Richard N. Taylor. Architectural styles and the design of network-based software architectures. University of California, Irvine Doctoral dissertation, 2000.*
35. *Singh, K., Guntuku, S. C., Thakur, A., &Hota, C. (2014). Big data analytics framework for peer-to-peer botnet detection using random forests. Information Sciences, 278, 488-497.*