Proceedings of the Third International Conference on Trends in Electronics and Informatics (ICOEI 2019)
IEEE Xplore Part Number: CFP19J32-ART; ISBN: 978-1-5386-9439-8

A Honeypot with Machine Learning based Detection Framework for defending IoT based Botnet DDoS Attacks

Ruchi Vishwakarma
Dept. of Computer Engineering
NIT Kurukshetra
Kurukshetra, India
ruchi.rbl@gmail.com

Ankit Kumar Jain
Dept. of Computer Engineering
NIT Kurukshetra
Kurukshetra, India
ankitjain@nitkkr.ac.in

***Abstract*— With the tremendous growth of IoT botnet DDoS attacks in recent years, IoT security has now become one of the most concerned topics in the field of network security. A lot of security approaches have been proposed in the area, but they still lack in terms of dealing with newer emerging variants of IoT malware, known as Zero-Day Attacks. In this paper, we present a honeypot-based approach which uses machine learning techniques for malware detection. The IoT honeypot generated data is used as a dataset for the effective and dynamic training of a machine learning model. The approach can be taken as a productive outset towards combatting Zero-Day DDoS Attacks which now has emerged as an open challenge in defending IoT against DDoS Attacks.**

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

I. INTRODUCTION

IoT which is a network of interconnected things without human intervention has also now become the source of propagating DDoS Attacks [1]. IoT devices can be more easily compromised than desktop computers. Therefore, there is a significant increase in the occurrence of IoT-based botnet attacks [7]. The botnet referred to as a network of bots (compromised IoT devices), is the result of malware infections in an IoT network [2]. According to the recent survey, there are over 6 billion IoT devices on the planet, such a huge number of potentially vulnerable gadgets cannot go easily unnoticed by cybercriminals. The thousands of malware are detected in previous years, and about half of them were in the year 2017 only [5].

A honeypot, as its name suggests, used for luring in attackers with an intention to observe and analyze their method of launching an attack by capturing information about the attacking agent like malware for a DDoS attack [9]. It is a device capable of getting compromised on the behalf of the main server by simulating any vulnerability which can easily be exploitable by an attacker. The information which it can capture by monitoring the activities between the attacker and itself are IP addresses as well as MAC address, port numbers, the type of devices targetted by it and also about the malware executables and its commands, etc [27]. In the past years in the field of computer security, honeypots have been proved out as a great source for researching out the various malware and its variants. It first came into existence in the late 1990s as 'The Deception Toolkit' which was developed by Fred Cohen in 1998 [28] and later became publically and commercially available especially to tackle with the self-replicating programs called worms.

Nowadays, there are different types of Honeypots available to be used by various applications. It can be classified depending on the level of interaction it allows with the attacker. The level of interaction depends upon the amount of data that needs to be get collected. Therefore, it is categorized into Low interaction honeypots and High interaction honeypots [9]. It can also be classified on the basis of objective it wants to attain i.e either they can be used for carrying out any research to get knowledge of possible threats and shortcomings in the system called as Research Honeypots, or they can be used for protecting the companies assets from the attacks in real time to improve the overall security called as Production Honeypots. Thus, honeypots are quite effective in dealing with Zero-Day DDoS Attacks without compromising IoT devices [29].

However, there is a difference between traditional honeypots and IoT honeypots. Traditional honeypots have similar architectures (mainly x86 and x86-84) whereas the architectures of IoT honeypots are heterogeneous due to different types of IoT devices.

In our proposed solution we have used a honeypot framework to catch several malware installation attempts into the IoT device. The collected information in the form of log files can be used as input to the machine learning model we are using for training purpose. The advantage of using honeypot over datasets to train the model is that we would be able to learn the model by unknown variants of malware families also instead of using only limited known data [13].

The concept of machine learning is used in our solution to automate the process of detection and prediction of the

incoming security threats to the IoT devices by using appropriate learning algorithm and techniques [17]. Learning algorithms are generally categorized into supervised and unsupervised. Supervised learning requires the assignment of labels of classification during the training process that can be used to predict the labels if corresponding features are relatively the same. On the other hand, unsupervised learning [6] does not require such labels to be assigned, in fact, it classifies on the basis of similarities among the various features of the training dataset. In our solution, we prefer to use unsupervised learning algorithm as we do not want human intervention in the process because an expert is needed to form the rules and assign the labels accordingly. Some of the most common unsupervised learning methods are clustering, anomaly detection, and neural networks. The malware detection can be characterized by a classification problem or a clusterization problem [10, 11]. Classification problem has known instances of data, hence uses supervised learning to predict the nature of the problem into classes. In the clusterization problem, unknown malware types are clusterized into several clusters based on the certain properties identified by an unsupervised learning algorithm [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].

II. Related Work

There are several honeypot based approaches are present in the literature for defending DDoS. The concept of the signature matching method had been used as a detection framework in some of these approaches [16]. Malware is detected on the basis of signatures obtained from their corresponding generated log files from the honeypot [18]. This type of detection was able to deal with only stored signatures and its variations, hence throw a limitation on dealing with an unknown and wider range of malware families. Another solution is anomaly based detection [12] which does not make use of rules, but a threshold is set for normal user behavior and any deviation from it leads to a declaration of possible malicious behavior. Such systems do suffer from high false positive rates because attackers now can imitate normal behavior too. Moreover, a machine learning based solution is capable to deal with such problem due to its ability to learn and teach over time. Thus, a more accurate classification with less number of false positive can be achieved by training the model with effective and updated data. The machine learning concept is used to better utilize the dynamic data produced by honeypot and increase the predictability for future attacks.

Many machine learning methods have also been proposed to identify DDoS based on the selection of statistical features using several supervised learning algorithms like SVM, Naïve-Bayes, etc [15,17]. However, these methods require extensive network expertise for selecting appropriate features out of the dataset and usually are limited to only one or several DDoS vectors. In addition, they require regular updates of the system to keep it functioning in diverse situations.

Another machine learning based solution was proposed to detect DDoS using deep learning models like: 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]. A network-based anomaly detection method was proposed which extracts behavior snapshots of the network and uses deep autoencoders to detect anomalous network traffic emanating from compromised IoT devices [26]. However, deep learning models need a large amount of data to train itself for producing accurate outcomes. In spite of that, they have extremely computationally expensive and complex training procedure and often require a significant amount of time to learn. IoT devices cannot afford such extensive procedures as they are quite constrained in terms of resources as well as in providing real-time services to the user. Moreover, there is a need to develop new methods for detecting attacks launched from compromised IoT devices and differentiate between hour and millisecond long IoT-based attacks [26].

III. Methodology

In our proposed solution, we are not only concentrated over detecting the malware but also interested in identifying the unknown malware families responsible for the category of Zero-Day DDoS attacks. Zero-Day attacks are caused by different possible variants of malware infections that yet not have been identified entirely for creating a complete DDoS defense against it [19]. This issue is solved by using a honeypot approach with a machine learning based detection framework. A honeypot is used to intentionally lure in attackers with the purpose to capture the malware properties and its style of invading the security of IoT devices by recording the whole information about it in log files [16]. In addition to it, a machine learning based detection framework is used next to predict the possibility of an abnormal activity based on the log files generated by the honeypot using a light weighted classification algorithm preferably an unsupervised one as it does not require any expert to classify the training tuples into a malicious one or a normal on [20].

The architecture for our proposed solution is as follow: The process starts when an attacker attempts to inject the malware through an open port (telnet port 23 or 2323) by logging into an IoT device using several combinations of ID and Passwords. Honeypot here comes into the picture for intentionally allowing to gain access to the attacker by invading its own protection wall. The main intention is to get the information about the malware as well as about the attacker by recording each activity between the device and the invader in the form of log files. These log files capture the information that enables us to get the idea about the nature of new malware families, their variants, type of targeted devices and also about the C&C server IP address, port number, etc. Now, since we have to train our machine learning model, we need to transform our log file data into a proper tabular format that will work as datasets. For classification, we will prefer to use a memory efficient which use minimum possible training data to predict the useful information, for avoiding it from becoming an overhead for an IoT device [20]. At last, based on the classification result, appropriate action is performed. Fig.1. represents the whole

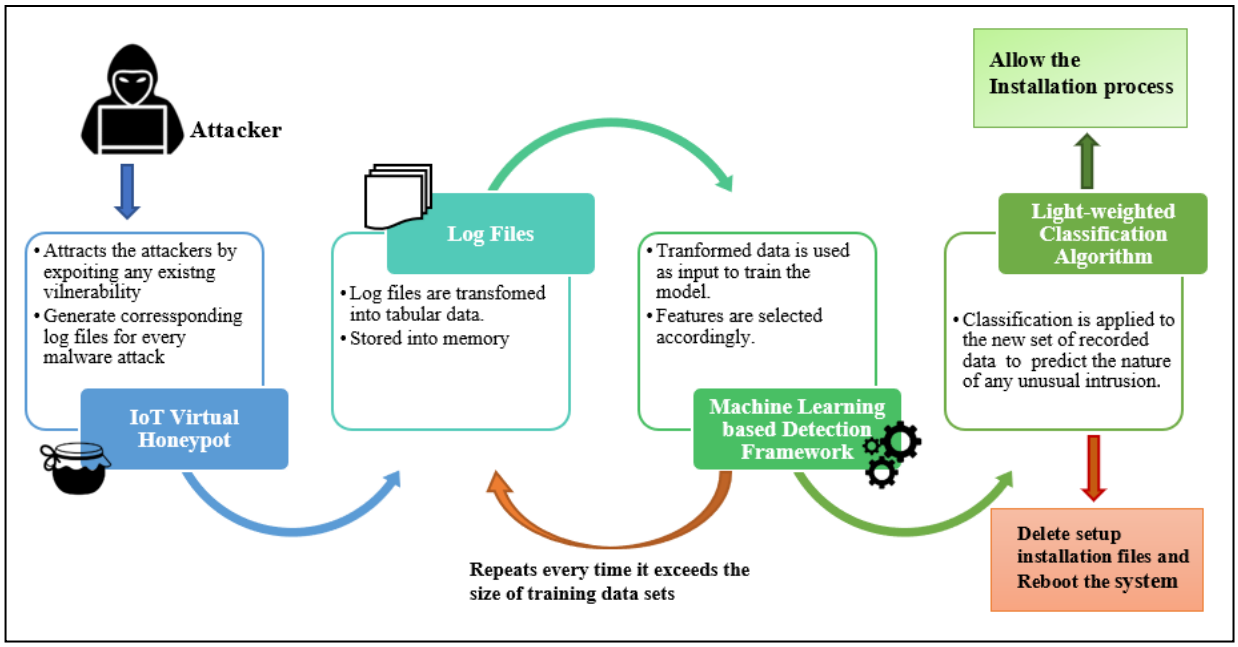

Fig. 1. Process flow for the honeypot-based solution with machine learning based detection framework.

process flow for the proposed solution. The process of training repeats every time it exceeds the allowable size of training data to make the process dynamic, and easily runnable on resource-constrained IoT devices.

IV. Implementation Aspects

Implementation is a necessary part of any novel approach or idea in order to check the feasibility and evaluate its efficiency over the currently available similar solutions. As discussed in the above section, our proposed approach consists of different subsequent steps. At each step, we can apply the latest methods for the underlying concept to keep our solution updated enough to handle the current IoT challenges. Following are the recent developments that took place in recent years in the field of IoT honeypots and real-time machine learning detection which are the two most important steps used in our approach for carrying out the desired implementation:

A. IoT Virtual Honeypot:

Our very first step in our proposed approach is to attract the attackers for deliberately exploiting the vulnerability present in IoT devices. For emulating this behavior, we need a system or device which can exactly act as an exploitable IoT device and prompt the attacker to play his malevolent move without having the second thought about the genuineness of the exploits. Such systems are widely known as IoT honeypots. As discussed above in the introduction based on the level of interaction, honeypots can be classified as High Interaction Honeypots (HIH), Low Interaction Honeypots (LIH) and Medium Interaction Honeypots (MIH) which is a combination of both. Since it's infeasible to set up a high interaction honeypot (HIH) for resource-constrained IoT devices, it would be preferable to select medium interaction honeypot (MIH) over the other two honeypots. That is the reason why it is named as IoT 'Virtual' honeypot as in this case we would be implementing it virtually by simulating the IoT platform using IoT communication protocols. The attack strategies like network traffic, payload, malware samples, the toolkit used by the attacker, etc. are then can be recorded by the honeypot. There is a list of some recently developed IoT honeypots for DDoS detection:

- IoTPOT [32]: This honeypot also emulates Telnet services of various IoT devices and consists of a frontend low interaction responder cooperating with a backend high interaction virtual environment called IoTBOX capable of operating at different CPU architectures.
- Telnet IoT honeypot [30]: Telnet server is used for implementing the trap for IoT.
- HoneyThing [31]: This honeypot emulates a vulnerable modem/router (having RomPager embedded web server) and is TR-069 (CPE WAN Management Protocol) specific.
- Dionaea [33]: This honeypot uses MQTT protocol to simulate the IoT behavior.
- ZigBee Honeypot [34]: This honeypot simulates 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 simulating an entire IoT platform, rather than a single application-layer communication protocol (e.g., Telnet, HTTP, etc.).

The most appropriate IoT honeypot should be able to emulate the IoT devices not by just emulating some selected IoT communication protocols, but it should be capable enough to simulate the whole IoT platform along with all the supported application layer protocols. Some of the most popular application protocol which is used for IoT communication are MQTT (Message Queue Telemetry Transport) by IBM, XMPP(Extensible Messaging and Presence Protocol) that provides basic instant messaging (IM) and presence

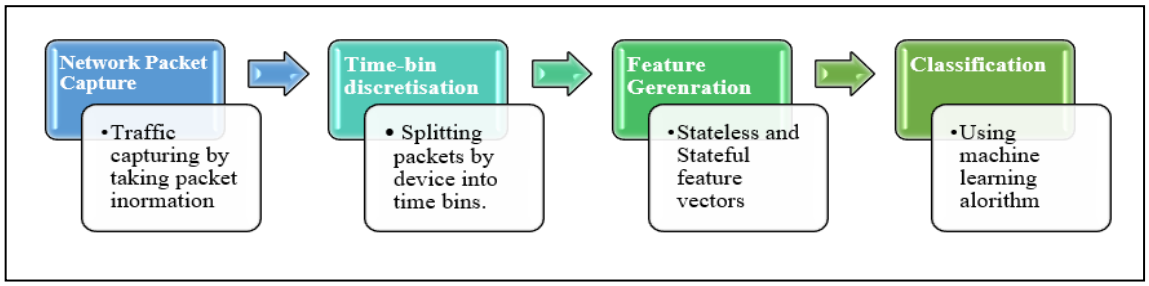

Fig. 2. Process flow for the machine learning based detection framework.

functionality, AMQP (Advanced Message Queuing Protocol) that arose from the financial industry, CoAP (Constrained Application Protocol) designed for resource-constrained devices., UPnP (Universal Plug and Play) set of network protocols used for the discovery of network devices and HTTP REST. REST is an architectural style that has been widely used in Machine-to-Machine (M2M) communications and IoT platforms. Among all the above-listed honeypots, we can use ThingPot for our purpose of an intriguing number of possible malware attacks.

B. Real Time Machine Learning Detection Framework

Machine learning based detection framework is another important step in our whole process of DDoS detection. There is a number of machine learning algorithms available for carrying out the desired classification. However, we are not interested in just the classification of malware, but we want a real-time implementable machine learning solution that can classify the malware features accurately without generating a number of false positives. The recent researches in the field of real-time machine learning based detection in IoT devices include a solution proposed by R. Doshi et al.,2018 [17] which has proved to classify the malware with an accuracy of 0.99. The solution is especially targeted to IoT botnet attacks that have shown a drastic increase in the past recent years.

IoT traffic behaves differently from that of traditional laptops and smartphones as the devices communicate with endpoints within a small range rather than large web servers. This kind of behavior of IoT traffic can be observed closely via a machine learning process. The process compromises of several steps starting from data collection, feature extraction, and then finally binary classification. The features extracted are mainly IoT-specific network behaviors and possesses network flow characteristics such as packet length, inter-packet intervals, and protocol. A variety of classifiers for attack detection, including random forests, K-nearest neighbors, support vector machines, decision trees, and neural networks are compared against each other. The random forest, K-nearest neighbors, and neural net classifiers were found to be particularly effective [17]. The IoT specific network behaviors like the limited number of endpoints, the regular time interval between packets, etc can be used to perform feature selection process to achieve the higher accuracy in detecting DDoS in IoT network traffic with the assistance of various machine learning algorithms, including neural networks.

Anomaly detection is the process that goes via different phases starting from *Traffic Capture*, then on *Grouping the packets by device and time*, and then coming on the *Feature Extraction* phase and finally ends on *Binary Classification* phase. The traffic capture process is about recording the source IP address, source port, destination IP address, destination port, packet size, and timestamp of all sent IP packets from IoT device that is a part of some smart home application. This task of collecting the DDoS traffic is a quite challenging task due to some involved security risks and complexity. It has simulated the three most common variations of DDoS attack i.e. a TCP SYN flood, a UDP flood, and an HTTP GET flood to capture the new coming variants in the malware properties.

Grouping is performed on packets from IoT devices based on source IP address which is further divided into nonoverlapping timestamps which were recorded at the earlier stage.

The feature extraction process is responsible for generating stateless and stateful features for each packet depending upon the IoT device behavior. *Stateless features* are lightweight features derived from flow independent characteristics of each sent packet i.e. they are generated without splitting the incoming traffic stream by IP source. On the other hand, stateful features are about capturing the aggregated flow information in the network traffic with respect to the short time spans. Packet size and Inter-packet interval are considered as stateless features whereas bandwidth and IP address cardinality and novelty are called stateful features. At last, binary classification is processed using different classification algorithms like K- nearest neighbors, random forests, support vector machines and deep neural networks to distinguish the normal traffic from the DDoS traffic flow [36]. Fig. 2 shows the complete flow of the processes involved. Moreover, using deep learning classifiers will be much effective as they work on the additional data generated from the real-world deployments

To summarize, the proposed solution can be implemented by using an IoT honeypot inspired by the ThingPot [29] which is an IoT virtual honeypot capable of catching various botnet binaries by emulating different IoT communication protocol along with entire IoT platform behaviors. To keep it isolated from the original IoT platform, the virtual box should be used to deploy it over every IoT device in a network. Since due to the IoT constraints, it is not possible to implement classifiers on each device, it should be implemented on the router level. Also, the amount of traffic coming to a particular IoT device is insufficient to perform any training over a machine learning

model. To generate an adequate amount of IoT traffic, any IoT network simulators can be used. IoT simulators are known for generating an IoT environment for testing any IoT-based application and add storage facility using cloud if required. However, if we are using the preferred honeypot, then there is no need to bother about IoT simulators as our honeypot itself be responsible for it. The transformation of log files into the format required for input to Machine learning model can be done by using bash scripts on Linux. For carrying out the machine learning tasks, machine learning tools like Microsoft Azure, MATLAB, etc. in a virtualized environment can be used. We can use the approach as discussed above for the real-time machine learning detection framework.

V. CONCLUSION

Internet-of-things is the biggest reason for the modernization of the real world in terms of technology. But it is also the main reason for the increasing number of cyber attacks especially DDoS attacks. That's why defending against such attacks that use IoT as a medium to harm network security has become the primary concern in the field of Internet Security. A number of defense mechanisms have been proposed in the concerned field to make the IoT network immune to such attacks but they become incapable of handling new variants of IoT botnet attacks. We came up with a honeypot based solution for the DDoS detection which uses real-time machine learning detection framework. Use of honeypots will ensure the logging of newly coming malware features which will be utilized by ML-based detection framework to train their classifiers effectively. For the future scope, we need to extend this approach to the next level where we can find out the open challenges or issues by implementing over the real-time scenarios. There is also scope for employing a cloud server to deal with extremely resource-constrained IoT devices. Finally, we can come up with a comparative analysis of our proposed solution by evaluating its performance in contrast to other proposed models.

REFERENCES

[1] K. Chen, S. Zhang, Z. Li,Yi Zhang, Q.Deng, Sandip Ray, Yier Jin, "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/A10-TPS-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 zero-day 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] A. 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-iot-honeypot.

[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.

[33] DinoTools. dionaea - catches bugs. https://github.com/DinoTools/dionaea/blob/master/README.md.

[34] 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.

[35] Roy T. Fielding and Richard N. Taylor. Architectural styles and the design of network-based software architectures. University of California, Irvine Doctoral dissertation, 2000.

[36] 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.