A Brief Review of Advanced Monitoring Mechanisms in Peer-to-Peer (P2P) Botnets

Mahmood A. Al-Shareeda
National Advanced IPv6 Center
Universiti Sains Malaysia
Penang, Malaysia
alshareeda022@gmail.com

Selvakumar Manickam
National Advanced IPv6 Center
Universiti Sains Malaysia
Penang, Malaysia
selva@usm.my

Murtaja Ali Saare
Computer Technology Engineering
Shatt Al-Arab University College
Basrah, Iraq
murtaja.a.s@sa-uc.edu.iq

Shankar Karuppayah
National Advanced IPv6 Center
Universiti Sains Malaysia
Penang, Malaysia
kshankar@usm.my

Murtadha A. Alazzawi
Computer Techniques Engineering
Imam Al-Kadhum College (IKC)
Baghdad, Iraq
murtadhaali@alkadhum-col.edu.iq

Abstract—Due to threats from botnets, internet security is becoming less safe. An attack plan can only be planned out to take down the botnet after the monitoring activities to understand the behavior of a botnet. Nowadays, the architecture of the botnet is developed using a Peer-to-Peer (P2P) connection, causing it to be harder to be monitored and tracked down. This paper is mainly about existing botnet monitoring tools. The aim of this manuscript is to investigate the ways to monitor a botnet and how the monitoring mechanism works. The monitoring tools are categorized into active and passive mechanisms. A crawler is an active mechanism, while sensor and Honeypot are the passive mechanisms. Previous work about each mechanism is present in this paper as well.

Keywords—Peer-to-Peer (P2P), Honeypot, Monitoring, Botnet Architecture.

I. INTRODUCTION

A botnet is a bot network. Infected computers that are used as controlled robots or machines for criminals, known as bot masters, to commit cybercrimes such as infecting computers and machines with viruses and malware, accessing multiple web pages concurrently to cause a denial of service (DOS), stealing sensitive personal information, spamming, and other cybercrimes are known as bots [1]–[3].

As shown in Fig. 1, when a user accesses a malicious email attachment, goes to a hacked website, or unknowingly downloads the bot files onto their machine, they become infected. Once infected, the botmaster will access the victim’s computer without the victim’s awareness [4]–[8]. By accessing to a victim’s computer, the botmaster will obtain two critical resources: CPU power and IP address [9]. Combining all the CPU resources from the bots can form a supercomputer service provided to the botmaster. The diversity of IP address will cause confusion and increases the difficulty of cyber-crime detection as all those IP addresses seem to be legitimate [10]–[16]. Industry estimates indicate that botnets have cost businesses worldwide over US$110 billion in losses [9]. The Internet security issues caused by botnets have even triggered the US government and American Industries’ awareness and listed it as the greatest threat to cyber security [17], [18]. IoT devices have been targeted as victims of cybercrime because of the increased speed of the Internet, and technological advancements [19]–[22]. As a result of IoT manufacturers frequently putting security at risk during different stages of the software development lifecycle, the problem became more critical [23]. In order to take down the botnet, long-term monitoring is required to be done before an attack strategy and method can be determined [24]. Botnet monitoring can be done through some monitoring mechanisms. These monitoring mechanisms are Crawler, Sensor, and Honeypot. Each of these tools is having different functionality in botnet monitoring activities [25]–[29]. Since botnet is an important asset for botmaster, so, botmaster has included some anti-monitoring mechanisms in their botnet design to prevent being tracked and removed threats from harming their botnet [30]–[32]. And these antimonitoring mechanism has increased the level of difficulty for monitoring tools to continuously monitor the botnet due to the existence of those interferences and disruption [33]–[35].

A brief description of the botnet monitoring activities is as follows. Honeypot is normally used in monitoring activities to obtain the bootstrap nodes (first node list) for a crawler to crawl through the botnet and the entrance point for the sensor to be inserted into. Once all the mechanism is in place, the monitoring activities might take up a few weeks or months. It is important to understand how the botnet works in the design of the architecture and mechanism. Then, an equivalent attack plan can be prepared accordingly. The
attack plan aims to bring down all the nodes in a botnet at once because even a few nodes left behind might provide chances for the botnet to recover after sometimes [31], [36].

The rest of this paper is structured as follows. Section II describes the background of the botnet architecture and mechanism. Section III shows the description of botnet monitoring mechanisms. Section IV discusses the advantages and disadvantages of these mechanisms. Lastly, Section V concludes this paper and discusses about possible future works.

II. BACKGROUND

During forming an attack plan, the network analyst needs to understand botnet behavior as to design a matching attack plan. Therefore, to understand botnet behavior, we have to know more about botnet architecture and its mechanism. In this section, the botnet architecture and mechanisms are described.

There are two types of botnet architecture: centralized and decentralized or known as Peer-to-Peer (P2P). Based on the command-and-control (C2) protocol, the way the bots communicate or connect to the server varies depending on the botnet architecture [37]. In early botnet design, botnet structure was designed using only centralized architecture. However, due to centralized architecture suffering a single point of failure, Peer-to-Peer (P2P) architecture is much preferred in botnet architecture design nowadays [38].

In P2P architecture, there are structured and unstructured categories. In contrast to centralized botnets, P2P structured botnets allow bots to connect with one another over the P2P protocol in order to update information about their neighboring peers. Then, the botnet connection is fixed and does not change simply. Gnutella and Kademia botnet [39], [40] are examples of P2P structured botnet. In P2P unstructured botnet, there are no fix formation for this type of botnet. Hence, it is very hard to be monitored. Sality, ZeroAccess, and Kelihos are some examples of P2P unstructured botnet that is still active in action till the year 2013 [41].

- **P2P Structured Botnet**: The Distributed Hash Table (DHT) decentralized distributed system, which offers lookup services via a hash table (key, value) pairs and stores to the DHT of every bot, is used to create a structured P2P botnet. The indexes are used to create the botnet connection [39], [40].

- **P2P Unstructured Botnet**: Unlike structured P2P botnets, unstructured botnets do not construct their structures utilizing the DHT protocol. By manipulating a new peer list or a bot’s neighbor list, the botmaster can change the construction of the unstructured botnet connection. This is done using the Membership Maintenance (MM) mechanism [32], [42].

- **Challenges in P2P Botnet Monitoring**: There are many challenges in P2P botnet monitoring activities. An antimonitoring mechanism designed in the botnet by the botmaster has always caused the most troublesome experience for botnet monitoring activities [30], [31]. Antimonitoring mechanism’s purpose is to prevent monitoring mechanisms like sensor and crawlers from being inserted into botnet networks.

III. BOTNET MONITORING MECHANISMS

There are three types of botnet monitoring tools: Honeypot, Crawler, and Sensor, as shown in Fig. 2. Honeypot and Sensor are passive mechanisms as these tools always wait for a connection from the bot. But, for the active mechanism Crawler, will act actively by probing the available nodes for the Neighbour List (NL) [43]. So that, from the list, it can identify the next node to probe.

![Botnet Monitoring Mechanisms](image)

**Fig. 2. Botnet Monitoring Mechanisms.**

A. **Honeypot**

A tool called a “honeypot” is used to capture bots from a botnet during an attack in order to collect information about their behaviors and analyze them to learn more about botnets [44]. It usually represents itself in the way of a server or high-value asset but is vulnerable in the system design. This is to make it become a potential target for network attackers and then gather information for network security researchers.

From the information, researchers can understand more about the techniques and mechanisms used in the botnet, but at the same time, they need to prevent the botnet from gaining access to the main systems [45].

Low interaction, medium interaction, and high interaction honeypots are the three different sorts of honeypot levels of interaction. The categories are established in accordance with the level of engagement or services offered by Honeypot to potential hackers.

- **Low Interaction Honeypot**: The risk of this Honeypot causing the whole server to compromise the botnet is low due to it is just a small pile from the server, and there is no operating system for the botnet to deal with [46], [47]. Some examples of the low interactive Honeypot are Honeyd, HoneyRJ [47], BotMiner, BotGrep and BotTrack [48].

- **Medium Interaction Honeypot**: Medium interactive Honeypot is better in terms of the information that it can obtain from the mechanism. That is because it is
designed to have more security holes so that bots from the botnet can access the system [47]. Hence, more information and complicated attacks from the botnet can be gathered compared to low interactive Honeypot. Mwcollect, Honeytrap, Nepenthes [44], HoneyBOT [47] and Kippo Honeypot Distro [49] are some of the medium interaction honeypots that are used today.

- High Interaction Honeypot: Unlike the other two, there is an operating system in this type of Honeypot. Hence, the botnet can perform any activities in this mechanism. Thus, more data can be gathered from the botnet activities through high interactive Honeypot. The most significant weakness of this Honeypot is that it is time-consuming and difficult to maintain [42], [44], [47], [49]. Specter and Argos are good examples of a high interaction honeypot [47], [49]. Besides that, there are Minos, ManTrap, and Argos [44].

**B. Crawler**

Crawler typically functions as an active mechanism in botnet monitoring software. It begins by probing the active bootstrap nodes and requesting NL from them. Once the NL is received, it will repeatedly ask all of the NL’s active nodes for new nodes until all nodes are found, or the action is canceled [42]. The crawling method is mostly developed utilizing the DFS approach. It starts off with first node in the bootstrap list and will stop after a parameter calculations. The crawler will start at the first node in the bootstrap list and will stop after a certain number of crawls. When all reachable nodes in the network were found or the limit was achieved, the crawling activities came to a stop.

| Honeypot Types | Advantages | Disadvantages |
|----------------|------------|---------------|
| Honeyd (Low Interactive) | Able to act like a normal operating system. Able to create and run with fake IP addresses simultaneously. | Aged and outdated NMAP fingerprint, which will expose the identity of the honeypot. The operating scripts are bound to different ports. |
| HoneyRJ (Low Interactive) | Extremely simple and easily extendable. Multi-threaded to support multiple connections. | Only offers protocols based on the string and does not offer binary data transmission. |
| Kippo (Medium Interactive) | Able to imitate other file systems and log all interactions with a nonautomated attack or automated attack. Utilized IoT device’s inerability as a lure to the hacker. | Only supports the partial Performance of services and do not permit full connection with the network. |
| Nepenthes (Medium Interactive) | Having Scalability and flexibility features. Able to create many honeypots in the system and capture data easily. | Many restrictions on the setting. |
| Honeywall (High Interactive) | Hard to be hacked by a hacker into Honeywell administration system. Able to capture more useful and interesting findings. | Implementation will be time consuming and complicated. Does not hide the network address. |

| Crawler Types | Advantages | Disadvantages |
|---------------|------------|---------------|
| P2P | Covered for all P2P botnet evaluation and operated in real-time | Not capable of handling anti-monitoring |
| Storm | BFS crawler, easy to design and crawl fast | Not capable of handling anti-monitoring mechanisms and churn effects |
| Nugache | DFS crawler, easy to design | Not capable of handling anti-monitoring mechanisms and churn effects |
C. Sensor

As soon as a sensor node is deployed in a botnet, it is passively waiting for connections from other bots. If the sensor node continues to present and interact with other bots, it will be promoted to a well-liked superpower and be able to join the bots’ preferred member list. The bots are going to start spreading information about this sensor node [54]. The messages regarding the active nodes in the botnet that are being sent back and forth between the botnets can then be observed. An analyst for network security will be able to identify the botnet nodes from these messages [42], [51]. This is the main purpose of a sensor which is to be as public as possible so that to be able to increase their visibility as higher as possible. Although sensors are able to obtain information about the bots, they cannot gather fine-grained information or get the interconnectivity of the bots [51]. The following subsection will describe some botnet monitoring research involving sensor nodes.

- Monitoring Kelihos Botnet: Starting from the bootstrap nodes in Kelihos botnet domains, they can track the growing the population of hosting IPs and detect new fast flux domains hosted by Kelihos botnet if the new domains match with the monitored DNS authoritative traffic. This will allow the author to analyze various components and attributes of the infrastructure used by the Kelihos fast flux botnet. During the monitoring, they also included a filtering function in order to avoid false positives result and confusion by other non-botnet fast-flux domains [18].

- Monitoring on Peer-to-Peer(P2P) Botnet: Based on Rossov et al. [41] observation, P2P botnets peers are periodically contacted by the neighboring peers, especially during regular peer list verification cycles. Besides that, the sensor can also be contacted by non-routable peers, unlike the crawler, which then makes sensor able to enumerate more nodes in a botnet. From the research results, sensor nodes have revealed a large number of bots that cannot be found by crawlers. Another methodological difference is that crawlers are actively enumerating peers, while sensors are working passively that it waits to be contacted by bots. And sensor’s coverage relies on its popularity in the botnet. So, the sensor node requires more time to leverage the outcome of the monitoring result.

- Passive P2P Monitoring in Storm Botnet: Passive P2P Monitoring (PPM) is a botnet monitoring tool designed by Kang et al. [40] that will act as peer nodes, primarily in Storm botnet’s P2P network. The sensor node will only listen in the Storm botnet and pretend itself as a legitimate bot and route messages. PPM might only have a small part of information about the whole network. It knows only those nodes in its routing table and those nodes that contacted it. This is because it only listens and will never contact other nodes by itself due to its passive nature. PPM can identify a node location regardless it is behind a NAT or a Firewall. Then, it will further distinguish the node into a different class. This is a function in which the sensor node overtakes crawler functionality [40].

### IV. ADVANTAGE AND DISADVANTAGE

Table I shows the comparison between different Honeybots according to their advantages and disadvantages described in different researches. Nowadays, Honeybots has utilized faster networking and visualization technologies during developing Honeybot. And, they have also deployed high-interaction Honeybots and low-risk Honeynets, isolating attack traffic from connected hardware and networks. It is good to see a lot of improvements that have been made in this mechanism to stop and prevent network security issues.

Table II shows the comparison between different types of crawlers in their strength and weakness. Most of the crawler seems lacking features to avoid anti-monitoring mechanism and churn effects. But, it might be due to some of the crawlers being designed long before the existence of those interferences. Future crawlers could need to contain characteristics that can withstand the effects of anti-monitoring mechanisms and take churn into consideration, like the crawler outlined in [41]. Because of the anti-monitoring mechanisms used by modern botnets, these qualities are crucial for a crawler’s effectiveness. Without them, the crawler’s performance could suffer, and the monitoring results could deteriorate, as stated in [32], [33], [35].

Table III shows the comparison between the different sensor in different botnet monitoring. Through the experiment result in [40], [41], it shows that more than 40% of bots that contact the sensor are behind firewall or NAT devices. Since the crawler is unable to reach those nodes, it is wise to combine crawlers and sensor nodes in monitoring activities to provide a much more accurate population estimation by using the advantage of a sensor to complement the weakness of the crawler [41]. It will help network analyst to obtain a better picture of a botnet connection.

### V. CONCLUSION

Despite the appearance that the three monitoring tools are enough for botnet monitoring. However, each of the monitoring tools has its flaws. Because it requires specific knowledge to implement, maintaining an efficient honeypot...
can be expensive for Honeypot. Otherwise, it could result in a configuration flaw that exposes the system to attack. There may come a time when the botnet will disable all of these monitoring programs. Therefore, if all these tools and methods cannot be improved or alternative mitigation techniques cannot be found quickly enough, there may be possible dangers from P2P botnet where all the existing ways may not be able to monitor or shut down the botnet in the future.

REFERENCES

[1] N. Failiere. Salty: Story of a peer-to-peer viral network. Rapport technique, Symantec Corporation, 32, 2011.
[2] S. Enterprise. Internet security threat report 2018. Mountain View, CA, USA, 2018.
[3] J. A. Wyke. The zeroaccess botnet-mining and fraud for massive financial gain. Sophos Technical Paper, 2, 2012.
[4] M. A. Al-Shareaa, Mohammed Anbar, Iznaz Husainy Hashullah, and Selvakumar Manickam. Survey of authentication and privacy schemes in vehicular ad hoc networks. IEEE Sensors Journal, 21(2):2422–2433, 2020.
[5] M. A. Al-Shareaa, M. Abar, Selvakumar Manickam, and Ali A Yassin. Vppes: Venet-based privacy-preserving communication scheme. IEEE Access, 8:150914–150926, 2020.
[6] M. Ali Saare, Azham Hussain, and Wong Seng Yue. Relationships between the older adult’s cognitive decline and quality of life: The mediating role of the assistive mobile health applications. 2019.
[7] M. A. Al-Shareaa, Mohammed Anbar, Murtadha A Alzawazi, Selvakumar Manickam, and Ahmed Shakir Al-Hiti. Lswbvm: A lightweight security without using batch verification method scheme for a vehicle ad hoc network. IEEE Access, 8:170507–170518, 2020.
[8] M. Mad Hamdi, L. Audah, Sami Abduljabbar Rashid, and Mahmood Al Shareeda. Techniques of early incident detection and traffic monitoring centre in vanets: A review. J. Commun., 15(12):896–904, 2020.
[9] N. Goodman. A survey of advances in botnet technologies. arXiv preprint arXiv:1702.01132, 2017.
[10] M. A. Al-Shareaa, M. Anbar, Iznaz H Hashullah, Selvakumar Manickam, Nibras Abdullah, and Mustafa Maad Hamdi. Review of prevention schemes for replay attack in vehicular ad hoc networks (vanets). In 2020 IEEE 3rd International Conference on Information Communication and Signal Processing (ICICSP), pages 394–398. IEEE, 2020.
[11] M. A. Al-Shareaa, Mohammed Anbar, Selvakumar Manickam, and Iznaz H Hashullah. Review of prevention schemes for man-in-the-middle (mitm) attack in vehicular ad hoc networks. International Journal of Engineering and Management Research, 10, 2020.
[12] M. Mad Hamdi, A. Shamal Mustafa, H. Falih Mahd, M. Salah Abood, Chanakya Kumar, and Mahmood A Alsharea. Performance analysis of qos in manet based on ieee 802.11 b. In 2020 IEEE international conference for innovation in technology (INCON), pages 1–5. IEEE, 2020.
[13] M. Ali Saare, Azman B Ta’a, Saima Anwar Lashari, and Sari Ali Sari. Mobile system for managing and mitigating the accommodation problems. In Journal of Physics: Conference Series, volume 1019, page 012045. IOP Publishing, 2018.
[14] M. A. Alazzawi, H.AH Al-behdalid, M. N Srayhi Almalki, Aqeel Luabi Challoob, and Mahmood A Al-shareaa. ld-ppa: robust identity-based privacy-preserving authentication scheme for a vehicular ad-hoc network. In International Conference on Advances in Cyber Security, pages 80–94. Springer, 2020.
[15] M. Saare, A. Hussain, and Wong Seng Yue. Investigating the effectiveness of mobile peer support to enhance the quality of life of older adults: A systematic literature review. 2019.
[16] M. A. Al-Shareaa, M. Anbar, Selvakumar Manickam, Aymen Khalil, and Iznaz Husainy Hashullah. Security and privacy schemes in vehicular ad-hoc network with identity-based cryptography approach: A survey. IEEE Access, 9:121522–121531, 2021.
[17] P. M Lutschker, N. B Weidmann, Margaret E Roberts, Mattjis Jonker, Alistair King, and Alberto Daimotti. At home and abroad: The use of denial-of-service attacks during elections in nondemocratic regimes. Journal of Conflict Resolution, 64(2-3):373–401, 2020.
[18] R. Marinho and Raimir Holanda. Exploring a p2p transient botnetfrom discovery to enumeration. The Journal on Cybercrime & Digital Investigations, 3(1):30–39, 2017.
[19] M. A. Al-Shareaa, M. Anbar, Selvakumar Manickam, Iz. H Hashullah, Nibras Abdullah, Mustafa Maad Hamdi, and Ahmed Shakir Al-Hiti. Ne-ppa: A new and efficient conditional privacy-preserving authentication scheme for vehicular ad hoc networks (vanets). Appl. Math, 14(6):1–10, 2020.
[20] A. Abdulatteef Abdulbari, M. Mohammed Jawad, HO Hanosh, Murtaja Ali Saare, Saima Anwar Lashari, Sari Ali Sari, Sarosh Ahmad, Yaser Khalil, and Yaqdhan Mahmood Hussain. Design compact microstrip patch antenna with t-shaped 5g application. Bulletin of Electrical Engineering and Informatics, 10(4):2072–2078, 2021.
[21] M. A. Al-Shareaa, M. Anbar, Selvakumar Manickam, and Iznaz H Hashullah. A secure pseudonym-based conditional privacypreservation authentication scheme in vehicular ad hoc networks. Sensors, 22(5):1696, 2022.
[22] MAASMAH Mannood A Al-Shareaa, Mohammed Anbar, Murtadha A Alzawazi, Selvakumar Manickam, and Iznaz H Hashullah. Security schemes based conditional privacy-preserving in vehicular ad hoc networks. Indonesian Journal of Electrical Engineering and Computer Science, 21(1), 2020.
[23] T. Sai Gopal, Mallesh Meenrolla, G Jyostna, P Reddy Lakshmi Eswari, and E Magesh. Mitigating mirai malware spreading in iot environment. In 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pages 2226–2230. IEEE, 2018.
[24] D. Stutzbach, R. Rejaie, and Subhrahna Sen. Characterizing unstructured overlay topologies in modern p2p file-sharing systems. IEEE/ACM Transactions on Networking, 16(2):267–280, 2008.
[25] Mahmood A Al-Shareaa, Selvakumar Manickam, Badiea Abdulkareem Mohamed, Zeyad Ghaele Al-Mekhlafi, Amjad Qtaish, Abdullah J Alzahrani, Gharbi Alshammari, Amer A Sallam, and Khalil Almekhaffi. Provably secure with efficient data sharing scheme for fifth-generation (5g)-enabled vehicular networks without road-side unit (rsu). Sustainability, 14(16):9961, 2022.
[26] M. A. Al-Shareaa, Selvakumar Manickam, Badiea Abdulkareem Mohamed, Zeyad Ghaele Al-Mekhlafi, Amjad Qtaish, Abdullah J Alzahrani, Gharbi Alshammari, Amer A Sallam, and Khalil Almekhaffi. Chebyshev polynomial-based scheme for resisting side-channel attacks in 5g-enabled vehicular networks. Applied Sciences, 12(12):5939, 2022.
[27] M. A. Al-Shareaa, S. Manickam, Badiea Abdulkareem Mohamed, Zeyad Ghaele Al-Mekhlafi, Amjad Qtaish, Abdullah J Alzahrani, Gharbi Alshammari, Amer A Sallam, and Khalil Almekhaffi. Chaotic map-based conditional privacy-preserving authentication scheme in 5g-enabled vehicular networks. Sensors, 22(13):5026, 2022.
[28] M. A. Al-Shareaa and S.R Manickam. Security methods in internet of vehicles. arXiv preprint arXiv:2207.05269, 2022.
[29] M. A. Al-Shareaa and S. Manickam. Man-in-the-middle attacks in mobile ad hoc networks (manets): Analysis and evaluation. Symmetry, 14(8):1543, 2022.
[30] D. Andriesse, Christian Rossov, Brett Stone-Gross, Daniel Plohmann, and Herbert Bos. Highly resilient peer-to-peer botnets are here: An analysis of gameover zeus. In 2013 8th International Conference on Malicious and Unwanted Software: “The Americas”(MALWARE), pages 116–123. IEEE, 2013.
[31] S. Karuppayah, S. Roos, Christian Rossov, Max Mulhìæ auer, and Mathias Fischer. Zeus muler: circumventing the p2p zeus neighbor list restriction mechanism. In 2015 IEEE 35th International Conference on Distributed Computing Systems, pages 619–629. IEEE, 2015.
[32] E. Vasilimonolakis, J. Helge Wolf, Leon Bock, Shankar Karuppayah, and Max Mulhiæ auer. I trust my zombies: A trust-enabled botnet. arXiv preprint arXiv:1712.03713, 2017.
[33] D. Andriesse, C. Rossov, and Herbert Bos. Reliable recon in adversarial peer-to-peer botnets. In Proceedings of the 2015 Internet Measurement Conference, pages 129–140, 2015.
[34] L. Bock, Emmanouil Vasilomanolakis, Max M’uhlh’user, and Shankar Karuppayah. Next generation p2p botnets: Monitoring under adverse conditions. In International Symposium on Research in Attacks, Intrusions, and Defenses, pages 511–531. Springer, 2018.

[35] S. Karuppayah, E. Vasilomanolakis, S. Haas, Max Muhl’ user, and Mathias Fischer. ‘Boobytrap: On autonomously detecting and characterizing crawlers in p2p botnets. In 2016 IEEE International Conference on Communications (ICC), pages 1–7. IEEE, 2016.

[36] D. Plohmann, Elmar Gerhards-Padilla, and Felix Leder. Botnets: Detection, measurement, disinfection & defence. European Network and Information Security Agency (ENISA), 1(1):1–153, 2011.

[37] Ping Wang, Sherri Sparks, and Cliff C Zou. An advanced hybrid peer-to-peer botnet. IEEE Transactions on Dependable and Secure Computing, 7(2):113–127, 2008.

[38] R. Singh Rawat, Emmanuel S Pilli, and Ramesh Chandra Joshi. Survey of peer-to-peer botnets and detection frameworks. Int. J. Netw. Secur., 20(3):547–557, 2018.

[39] T. Holz, Moritz Steiner, Frederic Dahl, Ernst W Biersack, Felix C Freiling, et al. Measurements and mitigation of peer-to-peer-based botnets: A case study on storm worm. Leet, 9(1):1–9, 2008.

[40] B. ByungHoon Kang, Eric Chan-Tin, Christopher P Lee, James Tyra, Hun Jeong Kang, Chris Nunnery, Zachariah Wadler, Greg Sinclair, Nicholas Hopper, David Dagon, et al. Towards complete node enumeration in a peer-to-peer botnet. In Proceedings of the 4th International Symposium on Information, Computer, and Communications Security, pages 23–34, 2009.

[41] C. Rossow, Dennis Andriesse, Tillmann Werner, Brett StoneGross, Daniel Plohmann, Christian J Dietrich, and Herbert Bos. Sok: P2pwned-modeling and evaluating the resilience of peer-to-peer botnets. In 2013 IEEE symposium on security and privacy, pages 97–111. IEEE, 2013.

[42] S. Karuppayah. Advanced Monitoring in P2P Botnets: A Dual Perspective. Springer, 2018.

[43] S. Karuppayah. Advanced monitoring in p2p botnets. 2016.

[44] D. Akkaya and Fabien Thalgott. Honeybots in Network Security: How to monitor and keep track of the newest cyber attacks by trapping hackers. LAP Lambert Academic Publishing, 2012.

[45] S. Lee, Azween Abdullah, and NZ Jhanjhi. A review on honeypot-based botnet detection models for smart factory. International Journal of Advanced Computer Science and Applications, 11(6), 2020.

[46] S. Lee, Azween Abdullah, Nz Jhanjhi, and Shi Kok. Classification of botnet attacks in iot smart factory using honeypot combined with machine learning. PeerJ Computer Science, 7:e350, 2021.

[47] E. Peter and Todd Schiller. A practical guide to honeypots. Washington University, 2011.

[48] F. Halta, Erkam Uzun, Necati S.is,eci, Abdulkadir Pos,ul, and Bakr Emre. An automated bot detection system through honeypots for largescale. In 2014 6th International Conference On Cyber Conflict (CyCon 2014), pages 255–270. IEEE, 2014.

[49] S. Dowling, Michael Schukat, and Hugh Melvin. A zigbee honeypot to assess iot cyberattack behaviour. In 2017 28th Irish signals and systems conference (ISSC), pages 1–6. IEEE, 2017.

[50] D. G Spampinato, Upasana Sridhar, and Tze Meng Low. Linear algebraic depth-first search. In Proceedings of the 6th ACM SIGPLAN International Workshop on Libraries, Languages and Compilers for Array Programming, pages 93–104, 2015.

[51] S. Haas, Shankar Karuppayah, Selvakumar Manickam, Max Muhlh’user, and Mathias Fischer. On the resilience of p2p-based botnet graphs. In 2016 IEEE Conference on Communications and Network Security (CNS), pages 225–233. IEEE, 2016.

[52] S. Karuppayah, M. Fischer, C. Rossow, and M. Muhlh’user. On advanced monitoring in resilient and unstructured p2p botnets. In 2014 IEEE International Conference on Communications (ICC), pages 871–877. IEEE, 2014.

[53] D. Dittrich and S. Dietrich. Discovery techniques for p2p botnets. Stevens Institute of Technology CS Technical Report, 4(26)/2, 2008.

[54] S. Karuppayah, L. Bock, T. Grube, Selvakumar Manickam, ‘Max Muhlh’user, and Mathias Fischer. ‘Sensorbuster: On identifying sensor nodes in p2p botnets. In Proceedings of the 12th International Conference on Availability, Reliability and Security, pages 1–6, 2017.