

Behaviour based botnet detection with traffic analysis and flow intervals at the host level

Sneha Padhiar, Ritesh Patel

U & P U. Patel Department of Computer Engineering, Charusat University, Gujarat, India

Article Info

Article history:

Received Jan 25, 2023

Revised Mar 3, 2023

Accepted Mar 23, 2023

Keywords:

Botmaster

Botnet

Centralized command and control

Host based detection

Traffic analysis

ABSTRACT

A botnet is one of the most dangerous forms of security issues. It infects unsecured computers and transmit malicious commands. By using botnet, the attacker can launch a variety of attacks, such as distributed denial of service (DDoS), data theft, and phishing. The botnet may contain a lot of infected hosts and its size is usually large. In this paper, we addressed the problem of botnet detection based on network's flows records and activities in the host. We proposed a host-based approach that detects a host, that has been compromised by observing the flow of in-out bound traffic. To prove the existence of command and control communication, we examine host network flow. Once the bot process has been identified in the host being monitored, this knowledge allows blocking any in/out traffic with the bot's server. In addition to providing information about the compromised machine's IP address and how it communicates with servers, the log file is generated, which can provide data about the command and control (C&C) servers. Most existing work on detecting botnet is based on flow-based traffic analysis by mining their communication patterns. Our work distinguishes itself from other methods of bot detection from its ability to use real-time host-related data for detection.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Sneha Padhiar

U & P U. Patel Department of Computer Engineering, Charusat University

Gujarat, India

Email: snehapadhiar.ce@charusat.ac.in

1. INTRODUCTION

Almost every daily task, including education, business, and entertainment, is handled via the network these days. This botnet term is a combination of robot and network. Robot are computer programs that are programmed to perform their tasks without any human interference [1], [2]. The bot can be good (e.g. chat bot) or bad. The bot which is part of the botnet is malicious program that has been installed on the victim's computer without his prior information. A bot owner takes control of the victim's computer and infects it with other bots in order to create a botnet of infected computers. The intruder is referred to as bot master. The bot master uses command and control server (C&C) to communicate with bots and transfer commands through the botnet [1]-[3]. Figure 1 shows the command and control flow of Bot master/Botnet life cycle as shown in Figure 1. Botnet life cycle consisting of 5 phases:

- Initial injection: In this step, the attacker identifies potential hosts by different methods. Once the attacker finds a suitable target, he/she uses a series of attacks to infect that host [3].
- Secondary injection: In this stage, the infected machine downloads the actual binary scripts of the malware [3].

- Connection: After infecting the computers, the botnet is still useless until the C&C server actually communicates with the bots [3]. Hence, after the two phases of infection, the C&C server makes contact with the bots to give command and control information on what is to be done by the bots.

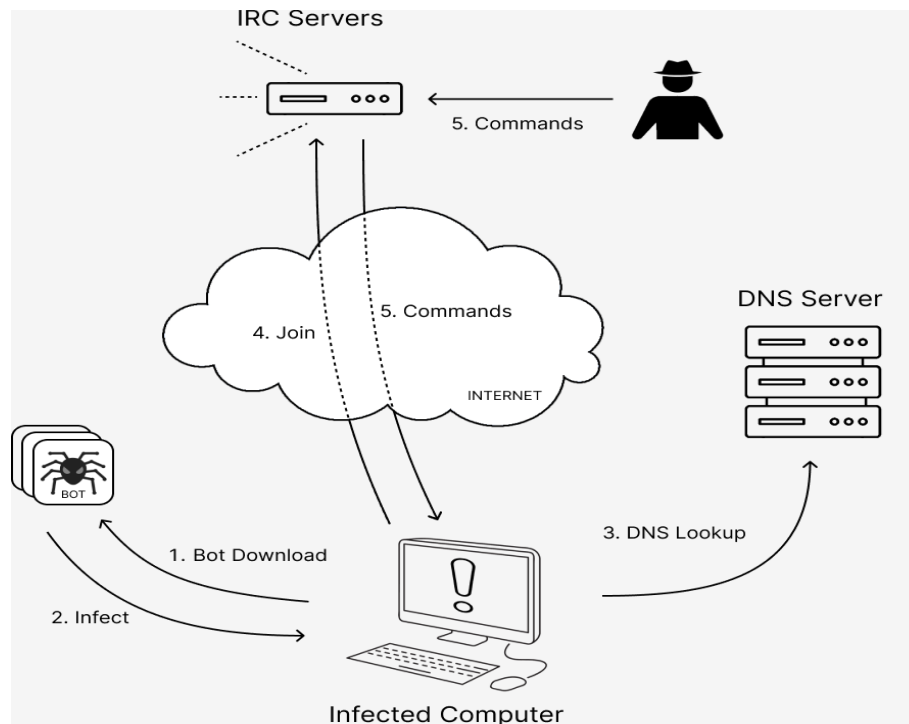


Figure 1. Botnet C&C architecture

- Command and control server: The heart of the botnets are the C&C servers. These servers are what the botmaster uses to communicate with and control the army of bots that was developed in the first 2 phases. With the help of the C&C server, the Controller will communicate commands to the slaves that are a part of the Botnet, and the C&C server will control all slaves/bots in accordance with the commands.
- Upgrade and maintenance: The most crucial step in software development is maintenance and up-gradation. Simply put, maintenance of a botnet is required to adapt it to new technologies making it more efficient, and easy to handle, and preventing C&C servers from being detected by anti-malware software and network analysts.

By manipulating the botmaster, the server can launch dangerous attacks such as click-fraud, data theft, spam, denial of service (DOS), distributed denial of service (DDOS), and phishing. A botnet's malicious objective is to gain access to a financial network for nefarious goal. They grab sensitive information from infected bias and send it back to the botnet's command and control server as part of one of the botnet's training. Furthermore, the botnet master leak this sensitive data [4]. Specific bank details or any other specific data that an unauthorized individual should not have access to are examples of sensitive information.

In recent years, botnets have become one of the biggest threats to the security of computers, being responsible for a great deal of malicious activity. The majority of internet users might be unaware. However, that their hosts have been compromised and are now part of a botnet [5]. Because newer botnets often use obfuscation techniques to evade antivirus scanners, these new botnets are harder to detect. Although corporation firewall is meant to allow genuine traffic, such as HTTP, peer to peer (P2P), and domain name server (DNS). This is taken as a benefit by the botmaster to bypass the corporation's firewall and install the bot into the user's machine [6]. Because of the large number of packets and data transfers in a network, it's difficult for network administrators to detect such an intrusion, since they cannot cover all the flows of information. Thus, we need to develop specific botnet detection technique to counter the botnets. Apart from centralized attacks against targets outside the network, botnets also compromise the host machine, installing backdoors and malware of various types. Anti-botnet counter measures currently rely primarily on monitoring and analyzing passive network traffic captured from switches and routers, looking for suspicious behaviour or signature based on similar network flow patterns. Whenever a match is found, the detection system issues a

warning. In order to suppress a botnet, the command and control server, which serves as a central point of communication between bots and botmaster, must be patched and killed [7].

With such amazing and sophisticated abilities. For normal user it is very tough to avoid or prevent infection by botnet. In addition, infection can be very damaging and that is why network-based detection approaches are insufficient. They cannot detect infected hosts, as well as notify them of infection. The tracing of botnet executables running on an infected computer, and the investigation of bots, is ineffective because network-based techniques are too slow to gather information useful for studying botnet behaviour. By stopping the C&C channel, botnets can be temporarily stopped. On the other hand, hosts will remain infected and compromised to the point that future attacks can exploit them. To set up a new botnet, botmaster usually use alternative addresses for their C&C servers. It is necessary to develop a real-time detection mechanism for botnet detection that can detect bots on the host, and to suppress the botnet and manage the disinfection process.

In this paper we proposed a new host-based approach that detects a host that has been compromised by observing the flow of inbound and outbound traffic. In order to prove the existence of C&C communication, we examine host network traffic. The rest of the paper is organized as follows. Previously related work is reviewed in section 2, the proposed detection approach and its components are described in section 3, and the procedure is concluded in section 4.

2. RELATED WORK

As botnets have become a powerful tool for obtaining confidential information from networks, and as a result of the risks these botnets can pose. Botnet detection in network has become a hot research topic. Various researchers in the literature specialize in the botnet detection technique. There have been several studies that analyze botnet behaviour based detection methods. Botnet approaches to detection based on behaviour can be categorized into 3 sections: i) host based detection, ii) network based detection, and iii) hybrid detection [8]. We are going to simply summaries some of the preceding studies that deal with these types of approaches.

2.1. Botnet detection at the host side

It is impossible to prevent an infected host from being infected by botnet malware by using firewall and antivirus software. Even after the C&C server is shut down, it's possible that diseased host can be used as a Launchpad for future attacks regardless of the shutdown of the C&C server. The bot program must be detected via a host based detection in this case to prevent it from infecting the host device.

Several researchers are working on a host-based detection technique. A solution proposed by Haung [9] for bot host- based detection is dependent on tracking network failures in a host over a limited period of time. Depending on the problem, two phases have been defined: i) training-phase and ii) detection-phase. The first stage uses failure flow features as data inputs, while the second stage uses knowledge derived during training to evaluate the data.

According to [10], Etemad and vahdani use host analysis in order to detect centralized C&C botnets. Their solution is based on the real-time analysis of traffic entering and leaving a host. It is made up of two basic elements: i) protocol classifier and ii) communication traffic pattern. Protocol classifiers separate outgoing and incoming traffic from the hosts first, then separate the infrared camera (IRC) and HTTP protocol traffic. After that, the separated traffic is forwarded to the communication pattern interpreter. This method determines if a given communication is legitimate or malicious through two modules: IRC and HTTP. The host firewall filters out these malicious packets. Detection of bots based on their communication with botnet command and control servers in IRC mode is conducted using the host's traffic analyzer. Detection of bots based on HTTP mode is accomplished by looking for periodic HTTP messages received from the botnet command and control server [11]. Neither the botnet master nor the bots process encrypted packets. Additionally, it only supports centralized botnets and is not compatible with P2P botnets.

As demonstrated Zeng *et al.* in [12], proposed an individual host-level preservation system. Each host-specific containment system includes two elements: i) behaviour analysis component and ii) containment model. Behavioral analysis is composed of a number of monitors and suspicion generators, that measure runtime behaviour of operating system processes including file system operations, registry operations and network layer functions. As a result of the process activity analysis, each process is assigned with a suspicious score. In the containment, the suspicious score is converted to a threshold based on the SVM algorithm. This approach offers several advantages: i) They emphasize behaviour analysis and containment, which is to be used for quick and automatic detection and containment of network worms, and ii) They use traces of real-world worm binaries and ordinary programs to do comprehensive analysis and testing.

2.2. Network-based botnet detection

Recent years have seen the emergence of methods using traffic analysis to discover botnets. In order to detect network traffic flow, several approaches have been proposed. Earlier botnet detection relied on payload methods that evaluated transmission control protocol (TCP) and user datagram protocol (UDP) packets for malware signatures. In addition to consuming a great deal of resources, payload techniques require processing many packets at a time and are slow. Moreover, newer botnets employ algorithms for encrypting and schemes to conceal their data transmission, as well as malicious payloads in crash packets.

Based on an analysis of flow of network traffic at regular intervals in [13] proposed a general botnet detection approach for detecting different types of botnets. Next, a static correlation analysis of network traffic flow is monitored in order to develop powerful classification model. Regardless of topology or protocol, it is possible to detect botnets using this approach. Additionally, it can detect unknown botnets. Using data mining algorithms [14] suggest a method for detecting P2P botnets based on monitoring and analyzing network traffic. They use three dataset analysis techniques: J48, naïve Bayes and Bayesian networks, which deliver 98%, 89%, and 87% accuracy, respectively.

Using network traffic analysis, Zhao *et al.* [15] suggest a new method for P2P detection of botnets that extracts behaviour for a predefined time period by selecting characteristics of the network flow. To differentiate botnet traffic from legitimate traffic, the investigators use the machine learning algorithm to extract decision trees. The investigators use the correlation attributes evaluator after choosing the decision tree to determine the most discriminating attribute for botnet detection. Their method able to spot offline bot activity and discover bots in early stages by detecting their activity in C&C phase. Furthermore, it has the ability to detect unknown bots as well as bots in early stages.

Hang and Sun [16] propose a machine learning-based botnet detection system that utilizes network traffic to detect botnets. The method starts with the selection and extraction of flow-based features from the network traffic. As a test, some noise is added to the payloads, inter arrival times, and features to ensure the model performs well. The proposed solution has an accuracy of 99.7% for some botnet types, which shows that it can handle more noise than an existing botnet detection solution.

According to Alauthaman *et al.* [17], their proposal offers a P2P anti-botnet detection approach dependent on decision-trees and multilayer neural networks with passive monitoring of network traffic. This framework identifies botnet communication between bots and the C&C servers. Next, 29 features are described in order to detect botnets. A feature reduction approach is applied to get rid of features that don't have much of an impact on the classification model to achieve high ranking of neural network learning and classification precision. This strategy's efficiency is demonstrated highly accurate performance with 99.0% and out performs most existing solutions. Certain solution focuses on botnet detection using DNS network traffic to detect botnets that rely on DNS to locate their C&C server to detect DNS detection techniques aren't useful for distinguishing C&C server traffic from new botnet types. Using iDns to identify domain names in the C&C server that are suspect, for an advanced persistent threat (APT) attack [18] propose reducing network traffic volume by deploying the system at the network edge. Their approach is composed of four components:

- Malicious DNS collector: Data collector detects suspicious APT C&C domains from the DNS record that are stored by the DNS records analyzer, then supplies the doubtful server IP address that corresponds to those domains to the another component.
- Data collector: In order to keep track of inward and outbound network traffic, a data collector is used.
- Component of network traffic analyzer: An anomaly detector and a signature detector are the two components of a network traffic analyzer.
- Reputation engine: Each IP address is assigned a reputation score, which is calculated from the previous components.

based on the analysis of 14 features, a malicious DNS detector can only detect malware that is reliant on DNS, including P2P botnets [19]. These features are categorized into four groups: i) domain name features, ii) time value features, iii) TTL value-based features, and iv) active probing features.

2.3. Hybrid based detection

Zeng *et al.* [20] propose a method of identifying botnets that integrates network-traffic analysis and host-traffic analysis. As the first solution to combine the detections at the host- level, network-level, and to correlate alerts, theirs has the potential to increase detection accuracy. A total 9 features are proposed for host-analysis: 6 features intended to analyze the task of file and registry operations, and 3 features that will analyze network traffic on the host. Network analysis is performed using 17 unique characteristics generated from Net flow data. They demonstrate the effectiveness of the combines host and network detection method against IRC, P2P and HTTP botnets.

Xu and Gu [21], it describes a technique for botnet detection that can monitor hosts as well as networks. To improve detection efficiency and effectiveness, an integrated multi-module approach is

developed using data from several sources of host and network-level perspectives [22]. A low false positive rate was observed with the effort platform, which was able to detect up to 15 bots.

A P2P botnet is discussed in [23] their solution depends on both host-based and network-based analysis. Host-level analysis looks for abnormal behaviour in the registry and log files. On the network-level, this analysis is focused on the full payload packet for every single event that occurred on the host. By doing this, you can distinguish between bot servers and bot hosts within the network. Besides the host analysis, it is effective at detecting P2P bots early on due to its ability to combine host analysis and network analysis. Using this method, the detection rate increases because it includes host analysis and network analysis. We mainly focus on the comparison of botnet detection techniques based on abnormal behaviour. The basic ideas, advantages and disadvantages of various methods are summarized in Table 1.

Table 1. Comparison of botnet detection technology methods based on abnormal behaviour [15]-[18]

	The basic idea	Advantage	Disadvantage
Deep learning	Based on temporal and spatial similarity, neural networks can identify network traffic features. Create a grayscale image of the network traffic and feed it into a neural network model that will detect and understand unique network traffic characteristics from the two dimensions of time and space.	<ul style="list-style-type: none"> – There is no need to know anything about the protocol and topology, it automates feature extraction, and it doesn't require any prior knowledge about the protocol. – In addition to detection capabilities against unknown botnets, it also has encryption protocol detection capabilities. 	<ul style="list-style-type: none"> – Using anti-machine learning techniques, attackers can escape. – Training speed for massive data is slightly slower.
Complex network	Zombie communication activities can be analyzed to form a correlation graph, and the complex network mining method is employed to decipher abnormal community behaviour based on the analysis of abnormal community behaviour.	Find invisible botnets with little or no traffic	When the dataset is large, the computation cost of the detection method is usually high, which affects the calculation of the behavioral association threshold.
Statistical analysis	Modelling zombie's behaviour and estimating its sample size based on statistical properties.	Quantitative and analytical analyses of statistical data can be conducted relatively quickly.	Statisticians have difficulty analyzing botnets due to their rapid change and complex features.
Distributed detection	Multiply the number of detectors in order to increase the flexibility of the detection system and collect massive amounts of data.	Improve accuracy The detection system should be more flexible.	<ul style="list-style-type: none"> – Comprehensive deployment strategies are difficult to choose. – Time consuming
Combinational method	<ul style="list-style-type: none"> – Multidimensional – Multiple technologies 	Allows detection of high-speed networks come across at an early stage	Lack of right aggregate can also additionally cause excessive computational cost

3. METHOD

Various researchers in the literature specialize in the botnet detection technique. As a device honor, we need to ensure that our device is not a part of any bot network. As a normal user, it is difficult to identify the normal and abnormal activity flows carried by their device. Ideally there should be some alerts that can notify device honor when some malicious activity takes place with high accuracy as part of the solution. As a result, a user can prevent according to their understanding and for the future they can take some precautionary measures so their device won't become part of a bot network. Here, we present a methodology for detecting and analyzing the bot in an infected host as well as perform an investigation on infected hosts.

3.1. Laboratory environment

This research takes place in an environment of physical and virtual devices. Two machines are utilized to collect malware and one machine to analyze malware activity. This lab consists of multiple computers. The cost of running this lab will be very high if only physical computers are used for analysis of botnets. However, the advantages of using virtual machines are that they reduce costs and can be restored to their original state in case the VMs are affected by a botnet. The implemented system is tested on a routed network infrastructure with a Windows-based attacking machine and Linux-based bot as shown in Figure 2. This saves time by eliminating the need to repeat the experiment numerous times, allowing us to achieve accurate and safe results [23].

3.2. Network analyzer

For monitoring the network, the system monitors the traffic of all devices, connected to the network and analyses it in order to observe connections patterns. We used the Wireshark tool for this purpose [24].

Table 2 shows the artifacts which are needed to monitor a network flow using wireshark tool. Given solution can detect the bot process by monitoring host processes involving registry and file systems, as well as network traffic. Table 3 shows characteristics for host process analysis. Selected characteristics are employed to determine a flow which includes address of host or a process in the host is likely to be a bot.

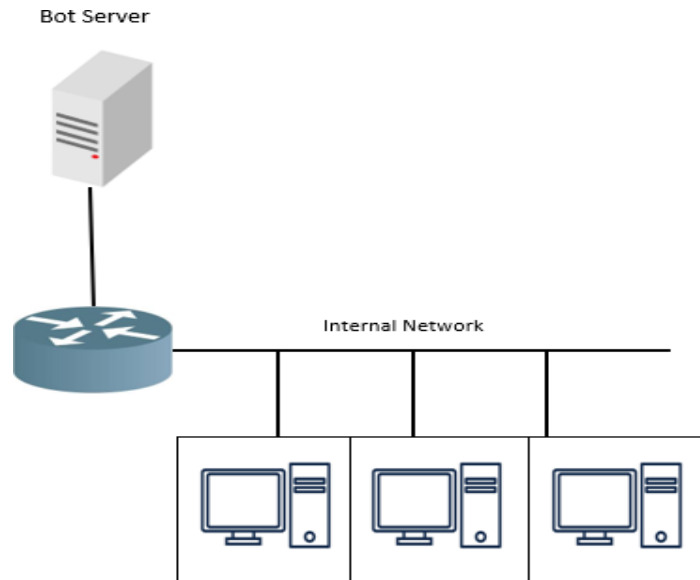


Figure 2. Experimental setup

Table 2. List of selected artifacts for network monitor

Number	Artifact
1	Port source and destination
2	IP source and destination
3	Protocol (UDP or TCP)
4	HTTP method (POST or GET)
5	Total number of connections
6	Total number of failed connections
7	Packet size

Table 3. List of selected artifacts for network monitor

Number	Artifact
1	Creation of DLL or EXE in system directory
2	Auto run key in registry
3	Active time of Bot process

Using real-time analysis of host traffic, we have proposed a detection method based on inferring C&C communication model existence from host inbound/outbound traffic [21]-[23]. The architecture of the proposed detection system is illustrated in Figure 3. Basically, there are two components: a protocol classifier and an interpreter of communication patterns. As a first step, the entire traffic being sent and received by a host is redirected to a component called protocol classifier. In this component, packets for IRC and HTTP are separated from the rest of traffic and passes them along to the next component. Using IRC part, communication pattern interpreter detects malicious traffic via IRC based on the bot’s communication model with C&C server. HTTP part, on the other hand, identifies HTTP-based botnet C&C communication patterns based on periodic repeatability of messages. A packet filtering firewall on a host can filter the output of a communication pattern interpreter as malicious traffic distinguish it from normal traffic.

3.3. Results and discussion on host behaviour analysis and identification of malicious activity

To identify the activity of botnets we have used active monitoring approach, which can highlight the pattern difference between the normal traffic flow and malicious traffic flow [25]. For that we have analyzed the overall utilization of CPU when our system is not infected by any malicious activity. Figure 4 Shows the

utilization of CPU when system is not infected. After that, we have observed behaviour of the device which is part of bot network. To achieve that multiple instructions were passed to host from bot master and analyzed the overall utilization of CPU. Figure 4 shows the utilization of CPU after the malicious activity. According to the literature review [25]-[27], whenever your device become a part of Bot network the Bot-master will boost the device and network traffic flow. After analyzing the literature review and performing multiple experiments on devices, we found that when a device is not infected, CPU utilization is between 20% to 40%, while when a device is a part of bot network and they are following orders, the CPU utilization is $\geq 70\%$. In some cases, the system will dangle.

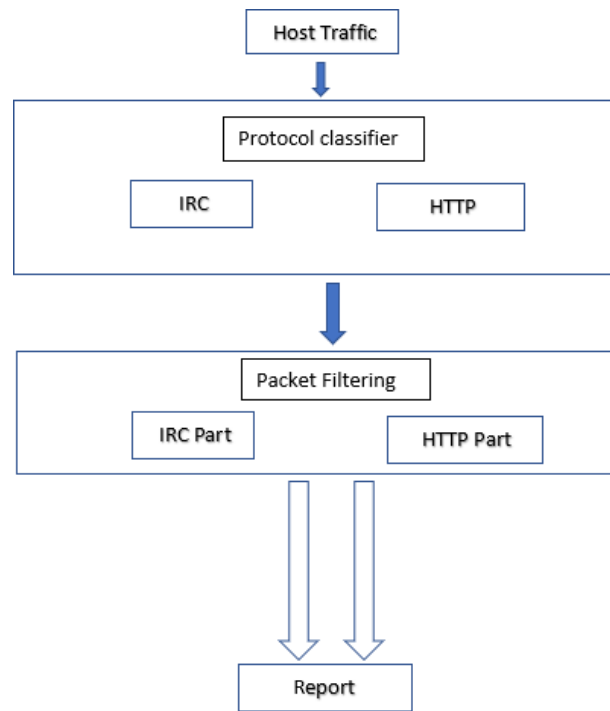


Figure 3. Architecture overview of our proposed work

By continuously monitoring CPU utilization and network traffic flow, it can capture the movement of the network and gather evidence of malicious activity. An alert will appear when malicious activity is detected, so device users can receive insight into some background malicious activity. Once the bot process has been identified in the host being monitored, this knowledge allows blocking any incoming or outgoing traffic with the bot's command and control server. In addition to providing information about the compromised machine's IP address and how it communicates with C&C servers, the log file can provide data about the C&C servers themselves.

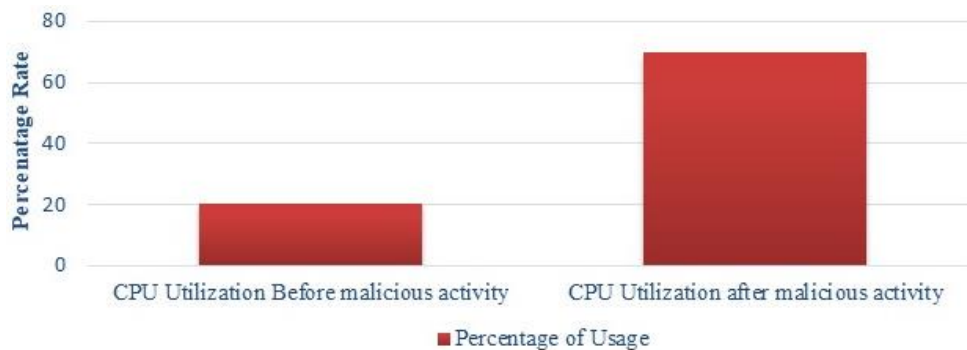


Figure 4. Utilization of CPU before and after malicious activity

4. CONCLUSION

A botnet is an advanced form of malware that is difficult to trace, detect, and stop, compared to other types of malware. Network-based techniques make it difficult to tracing the botnet executables on the infected machine and collecting evidence on the bots, which is helpful in studying the behaviour of botnets. By simply shutting down the command and control channel of the botnet, you can temporarily solve the problem; Despite this, hosts will remain infected and compromised, making them susceptible to future attacks. A limited number of works have been done to actively detect and block traffic from botnets on the infected host. A real-time detection mechanism is necessary to improve the previous botnet detection works. This mechanism would detect bots on the host for monitoring and controlling the disinfection process as well as filtering out malicious traffic to suppress botnets. In this paper, we addressed the problem of botnet detection based on network's flows records and activities in the host. We proposed a new host-based approach that detects a host that has been compromised by observing the flow of inbound and outbound traffic. In order to prove the existence of C&C communication, we examine host network traffic. Once the bot process has been identified in the host being monitored, this knowledge allows blocking any incoming or outgoing traffic with the bot's command and control server. Also, our work here is limited to centralized Botnet C&C models. In the future, we will apply our Bot detection solution for real time data and try to perform classification on server-less infrastructure. We will also test the proposed model with larger datasets like CTU-13, ISCX and analyze the detection accuracy.





REFERENCES

- [1] P. Ohri and S.-D. S. G. Neogi, "Networking security challenges and solutions: A comprehensive survey," *International Journal of Computing and Digital Systems*, vol. 12, no. 1, pp. 383-400, Jul. 2022.
- [2] Y. Xing and H. Zhao, "Survey on botnet detection techniques: classification, methods, and evaluation, Hindawi," *Mathematical Problems in Engineering*, pp. 1-24, 2021, doi: 10.1155/2021/6640499.
- [3] S. A. Chaturved and L. Purohit, "Spam message detection: A review," *International Journal of Computing and Digital Systems*, vol. 12, no. 1, pp. 439-451, Aug. 2022, doi: 10.12785/ijcds/120135.
- [4] S. Almutairi and S. Mahfoudhand, "Hybrid botnet detection based on host and network analysis Hindawi," *Journal of Computer Network and Communications*, 2021, doi: 10.1155/2020/9024726.
- [5] S. Malik, K. Kannorpatti, and S. Azam, "Critical feature selection for machine learning approaches to detect ransomware," *International Journal of Computing and Digital Systems*, vol. 11, no. 1, pp. 1167-1176, Mar. 2022, doi: 10.12785/ijcds/110195.
- [6] G. Vormayr and J. Febini, "Botnet communication patterns," *IEEE Communications Surveys and Tutorials*, vol. 19, no. 4, pp. 2768-2796, 2017, doi: 10.1109/COMST.2017.2749442.
- [7] P. Wang, S. Spark, and C. Zou, "An advanced hybrid peer-to-peer botnet," *IEEE Transactions on Dependable and Secure Computing*, vol. 7, no. 2, pp. 113-127, 2010, doi: 10.1109/TDSC.2008.35.
- [8] M. Yahyazade and M. Abadi, "BotCatch: Botnet Detection based on coordinated group activities of compromised hosts," 7th *International Symposium on Telecommunications*, IEEE, 2014, doi: 10.1109/ISTEL.2014.7000838.
- [9] C.-Y. Huang, "Effective bot host detection based on network Failure models," *Computer Network*, vol. 57, no. 2, pp. 514-525, 2013, doi: 10.1016/j.comnet.2012.07.018.
- [10] F. Etemad and P. Vahdani, "Real-time botnet command and control characterization at the host level," in *Proceedings of the Sixth International Symposium on Telecommunications*, Tehran, Iran, Nov. 2012, pp. 1005-1009, doi: 10.1109/ISTEL.2012.6483133.
- [11] M. Rostami, B. Shanmugam, and N. Idris, "Analysis and detection of P2P botnet connections based on node behaviour," in *Proceeding 2011 World Congress on Information and Communication Technologie*, Mumbai, India, 2011, pp. 928-933, doi: 10.1109/WICT.2011.6141372.
- [12] Y. Zeng, H. Wang, G. Shin, and A. Bose, "Containment of network worms via per-process rate-limiting," in *Proceeding 4th International Conference on Security and Privacy in Communication Networks*, Istanbul, Turkey, Sept. 2008, pp. 1-10, doi: 10.1145/1460877.1460895.
- [13] G. Kirubavathi and R. Anitha, "Botnet detection via mining of traffic flow characteristics," *Computers and Electrical Engineering*, vol. 50, pp. 91-101, 2016, doi: 10.1016/j.compeleceng.2016.01.012.
- [14] W. H. Liao and C. C. Chang, "Peer to peer botnet detection using data mining scheme," in *Proceeding International Conference on Internet Technology and Applications*, Wuhan, China, Aug. 2010, pp. 1-4, doi: 10.1109/ITAPP.2010.5566407.
- [15] D. Zhao, I. Traore, and B. Sayed, "Botnet detection based on traffic behavior analysis and flow intervals," *Computers and Security*, vol. 39, pp. 2-16, 2013, doi: 10.1016/j.cose.2013.04.007.
- [16] C. Hung and H. Sun, "A botnet detection system based on machine-learning using flow-based features," *Proceedings of the SECURWARE*, 2018.
- [17] M. Alauthaman, N. Aslam, L. Zhang, R. Alasem, and M. A. Hossain, "A P2P Botnet detection scheme based on decision tree and adaptive multilayer neural networks," *Neural Computing and Applications*, vol. 29, no. 11, pp. 991-1004, 2018, doi: 10.1007/s00521-016-2564-5. Available at: <https://link.springer.com/article/10.1007/s00521-016->.
- [18] G. Zhao, N. Aslam, L. Zhang, R. Alasem, and M. A. Hossain, "Detecting APT malware infections based on malicious DNS and traffic analysis," *IEEE Access*, vol. 3, pp. 1132-1142, 2015, doi: 10.1109/ACCESS.2015.2458581.
- [19] A. K. Suborna, S. Saha, C. Roy, S. Sarkar, and M. T. H. Siddique, "An approach to improve the accuracy of detecting spam in online reviews," in *International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*, vol. 2021, 2021, pp. 296-299, doi: 10.1109/ICICT4SD50815.2021.9396881.
- [20] Y. Zeng, S. Saha, C. Roy, S. Sarkar, and M. T. H. Siddique, "Detection of botnets using combined host- and network-level information," in *Proceeding 2010 IEEE IFIP International Conference on Dependable Systems and Networks (DSN)*, Chicago, IL, USA, Jun. 2010, pp. 291-300.
- [21] S. S. Xu and G. Gu, "EFFORT: Efficient and effective bot malware detection," in *Proceeding 31st Annual IEEE Conference on Computer Communications (Infocom '12) Mini-Conference*, Orlando, FL, USA, Mar. 2012, pp. 71-80.





- [22] M. Sethi, "Email spam detection using machine learning and neural networks," *International Research Journal of Engineering and Technology*, vol. 08, no. 04, Apr. 2021, doi: 10.1007/978-981-16-5157-1_22.
- [23] R. Abdullah, M. Faizal, and Z. Noh "Tracing the P2P botnets behaviors via hybrid analysis approach," *European Journal Scientific Research*, vol. 118, no. 1, pp. 75-85, 2014.
- [24] A. Kapre and B. padmavathi, "Behaviour based botnet detection with traffic analysis and flow intervals using PSO and SVM," *International Conference on Intelligent Computing and Control Systems (ICICCS)*, IEEE, 2017, doi: 10.1109/ICCONS.2017.8250557.
- [25] R. Bapat *et al.*, "Identifying malicious botnet traffic using logistic regression," *In 2018 Systems and Information Engineering Design Symposium (SIEDS)*, IEEE, 2018, doi: 10.1109/SIEDS.2018.8374749.
- [26] S. Lysenko, K. Bobrovnikova, and O. Savenko, "A botnet detection approach based on the clonal selection algorithm," *International Conference on Dependable Systems, Services and Technology*, IEEE, 2018, doi: 10.1109/DESSERT.2018.8409171.
- [27] H. Dhayal and J. Kumar, "Botnet and p2p botnet detection strategies: a review," *In 2018 International Conference on Communication and Signal Processing (ICCSP)*, pp. 1077-1082. IEEE, 2018, doi: 10.1109/ICCSP.2018.8524529.

BIOGRAPHIES OF AUTHORS



Sneha Padhiar     is an Assistant professor in U & P U. Patel Department of Computer Engineering, Charusat University, Gujarat, India. She received her B.E.C.E. From Gujarat Technological University in 2014 and M.E.C.E. From Gujarat Technological University in 2016. Currently, she is pursuing doctoral course in Computer Engineering at CHARUSAT. Her major area of research includes information security and IOT. She can be contacted at email: snehapadhiar.ce@charusat.ac.in.



Dr. Ritesh Patel     is working as a Professor at U & P.U Patel Computer engineering department of Charusat University of Science and Technology, Gujarat, India. He has received his doctorate degree in 2017 from Charusat University His areas of interest include cloud computing, internet of things, communication and networking, computer architecture, software engineering and cluster computing. He can be contacted at email: riteshpatel.ce@charusat.ac.in.