International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I5P220

Volume 70 Issue 5, 185-193, May 2022

Original Article

A Signature-Based Botnet (Emotet) Detection Mechanism

Foram Suthar[1], Nimisha Patel[2], Samarat V.O. Khanna[3]

[1,2,3]*Indus University, Ahmedabad, India*

[1]fss01@ganpatuniversity.ac.in

Received: 11 February 2022 Revised: 18 April 2022 Accepted: 23 April 2022 Published: 22 May 2022

Abstract *- The Internet has become an essential part of life, especially after the COVID-19 pandemic. The increasing use of technology brings new challenges. Cyber security has emerged as a major threat during the pandemic. Distributed Denial of Service Attack (DDoS) attacks have become more refined than other cyber-attacks during the pandemic. The most important question comes into mind: What is the source of the DDoS attack? The answer is botnet which provides the platform for the attacker. A botnet has targeted the escalation of vulnerable systems. Therefore, real-life and accurate botnet detection and prevention techniques must be effectively designed. Due to this organized dataset, IoCs are required for a most dangerous botnet to prevent networks at an early stage. Various malware datasets have been published for the research work, but most are outdated. The author has proposed a new dataset of windows based botnets using different analysis techniques. This work provides the geolocation of the live malicious connection made by emotet. They have also presented the mechanism which calculates the IP reputation and detects botnet based on IoCs using snort Intrusion Detection.*

Keywords *- Botnet, emotet malware, Snort, Intrusion detection system, Intrusion prevention system, DDoS.*

1. Introduction

The coronavirus pandemic has created new challenges for many organizations to adopt a new 'work from home model. In the era of digital working, people are connected virtually. As per the cyber security experts, more than 4,000 cyber-attacks have been reported a day since the COVID-19 pandemic [1]. This new trend of cyber-attack through malware in a pandemic is called 'Fearware' [2]. Ministry of Home Affairs (MHA) said that more than 3000 new websites related to covid-19 spread fake and dangerous content [2]. The hackers are now designing several new computer viruses and malware relating to COVID-19 updates. The year 2021 was a terrible year for cybersecurity as many governments suffered from malware attacks like distributed denial of services (DDoS) a ransomware attack. Any program which purposely executes malicious code on victim machines (Smartphones, Computer, Internet of Things (IoT) devices, Computer networks) is considered malware. Criminals generally use malicious software to launch a cyber-attack on the victim's machines [3].

Malware comes with different categorizations, e.g., viruses, worms, key loggers, Trojans, ransomware, botnet, spyware, etc. Various dimensions or taxonomy systematically categorize the malware [4]. Attackers use these characteristics very smartly to design malware to perform dangerous cyber-attacks. As a cyber-security analyst, it is really important to deeply understand the taxonomy of malware. Table.1 shows the different taxonomy which differentiates the malware.

Table 1. Taxonomy to classify the malware types

Malware/Classification	Standalone/ Host-program	Persistent/Transient	Auto-spreading	Dynamically updatable
Virus	Host-program	Persistent	Yes	No
Worm	Standalone	Persistent	Yes	Yes
Trojan	Standalone	Persistent	No	Yes
Ransomware	Host-program	Persistent	Yes	Yes
Key logger	Standalone	Transient	No	Yes
Botnet	Both	Persistent	Yes	Yes
Memory Resident	Standalone	Transient	Yes	Yes

To protect the network or host against malware attacks, various prevention mechanisms like Intrusion Detection (IDS) have been implemented. IDS can be classified as

network-based intrusion detection (NIDS) and host-based intrusion detection (HIDS). Both can be classified as signature-based intrusion detection (SIDS), anomaly-based, and hybrid-based detection. Malware detection is defined as 'Collection, Examination, Analysis (static and dynamic)' of Indicator of Compromised (IoCs). During this process, detection mechanisms are applied to data to conclude the pattern, third-party connections, methods used, and which file and registry they affected. Since the detection process is linked with analyzing big data to design effective techniques in law enforcement [5], preparing a dataset of botnets requires critical analysis. Malware authors use various packing and encryption techniques to hide code as they know the static analysis process. Packed code cannot be statically analyzed. So it is difficult to identify the behavior of the malware. In dynamic analysis, code has been executed in a live environment, so it is easy to collect the all run time features of the malware. So both static and dynamic analysis are equally important to understand the malware's characteristics, which will help prepare a dataset of IoCs.

This paper provides the latest analysis of windows based botnet malware with a review of significant research work.

The main contribution is as follows:

- Analyze the impact of the windows-based botnet attack during COVID-19.
- Prepared a new realistic DDoSBot dataset of emotet and give a detailed description of the features.
- Implemented snorts-based SIDS to detect emotet botnet.

The remaining paper is systematized as follows: Section 2 shows the research work done in this domain, including the analysis of the existing botnet dataset. The background of the botnet and proposed mechanism is presented in section 3. In Section 4, the conclusion of this research is given.

2. Literature Review

The section covers reviews of journals, research articles, and conferences on various approaches to designing botnet prevention mechanisms. Many IDS have been classified for botnet detection on a different target. Two datasets have been broadly used for IDS.

2.1. DARPA Dataset

Defense Advanced Research Projects Agency (DARPA) dataset was created by MIT Lincoln LAB in 1998. One year later, the team published an updated dataset with some improvements suggested by the computer security community. The training dataset includes seven weeks of a network-based attack, and the testing dataset includes the two-week network-based attack. Outside sniffing data and inside sniffing data have been extracted in tcpdump format. The mentioned dataset can be considered outdated as it was created in 1998 [6].

2.2. KDD99

IDS mostly uses KDD99 as the dataset. This dataset is made of 42 features. It comprises 24 different attack types in training and 14 more attack types in testing [7]. These new 14 attacks test the capability of the IDS against unknown attacks. The output classification of KDD99 is divided into DoS, Probe, R2L (Root 2 Local), U2R (User 2 Remote), and normal [7]. The data of R2L and U2R are very less. KDD99 dataset contains duplicate records in training and testing, which generate a false alarm. This dataset is again outdated because it was created in 1999 by DARPA. Later on, the researchers created an improved dataset, "NSL-KDD," that reduced the redundancy compared to KDD99, KDD98, and DARPA.

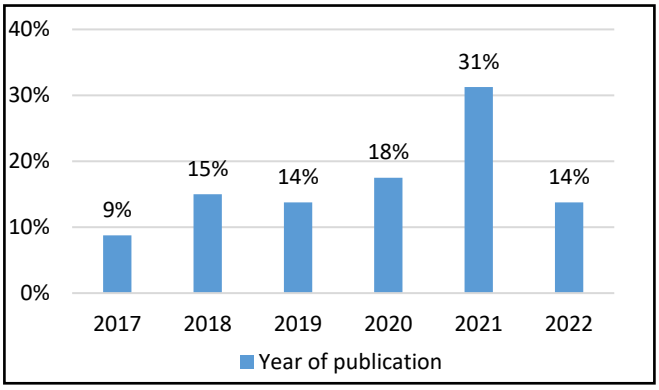
Another scientist also has created a novel dataset. The center of cybersecurity in Australia has created the UNSW-NB15 dataset. Coburg University creates the CICIDS dataset. Few recent datasets are in Table 2, chronologically listed with a short description.

Table 2. The most current datasets for the NIDP

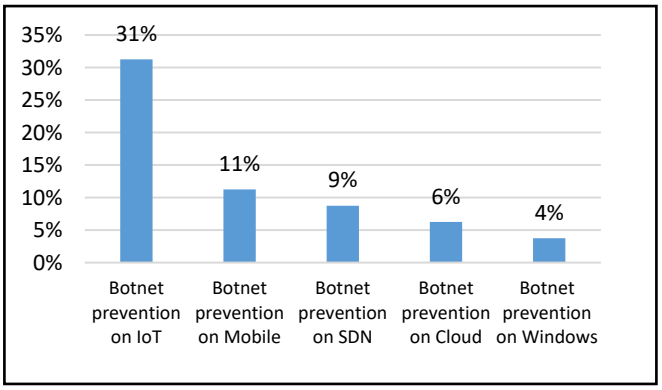
Sr.No	Dataset Name	Disadvantages
1	Defense Advanced Research Projects Agency (DARPA)	Outdated-created in 1998
		Dataset presents statistical issues and does not help detect novel malware
2	KDD99	Classified into U2R, Denial of Service, Probing, and R2L
		The most used dataset for IDS and ML but outdated - it was fabricated in 1999 by the DARPA
3	PNSL-KDD	The updated version of KDD and DARPA
		PNSL-KDD datasets to identify DoS attacks but generated in 2001
4	CICIDS 2017 - 2018	The size of the dataset is very hug and contains many redundant records.
		Contain IoT botnet only (executed through Ares botnet)
5	UGRansome	Latest dataset, designed in 2021
		Only design for ransomware

More than 80 articles related to botnet attack on different target has been reviewed. Figures 1(a) and (b) show the number of articles by item and year. As per the analysis, most of the research work has been done on IoT, SDN, and Mobile based botnets.

Summary of paper [8] highlights IoT attack literature and detailed static analysis. The review emphasizes the datasets utilized, research done focusing on primary selected studies, network forensic methods utilized, and covers existing proposed contributions. Although, the specific detection technology is not introduced along with the comparison and analysis of detection methods. Botnet detection technologies that are DNS-based can be broadly categorized into DGA-based, Flow-based, flux-based, anomaly-based, and bot infection-based. Here, the suggestion of a smart DNS-based botnet detection system with essential attributes has been highlighted. However, the survey does not cover the context behind the botnet's construction mechanism [9].



(a) Articles published per year



(b) Article s published per items

Fig 1. Number of articles publication details on botnet

A comprehensive analysis of botnet detection is done in [10]. Here, the survey's classification of botnet techniques is done in four ways: signature-based, anomaly-based, DNS-based, and mining-based. Regrettably, the summary excludes the latest technology. When it comes to the classification of DNS traffic analysis-based botnet detection technologies, there are two options [11]: Honeypot based and IDS based. These focus on decision trees, neural networks, clustering, statistical analysis, and graph theory. Although the literature is comprehensive, the merits of the same are not evaluated. [12] focuses on the evasion and detection techniques of DNS- based botnets. The introduction is made for the DGA botnet detection technique and Fast-flux. Also, this survey does not have any evaluation, and the dimensions are relatively single.

A unique botnet detection method which is a hybrid, is based on HANANBot on the host and network analysis was highlighted in this paper [13]. This technique detects new botnets in early-stage using NB and DT algorithms. Shortfalls of this research include updating rules dynamically or configuring files or signatures is still challenging. Paper [14] proposes a fresh botnet defense mechanism built on network strategy and Honeypot. The author has used MTD: reinforcement learning algorithm to detect the botnet. The issue in this technique is high time complexity. The author has designed the mechanism to detect the internet of thing botnet in [14] paper. The author has used ML algorithms like SVM, RF, Bagging, DT, and KNN and applied the techniques to the IoTPOT dataset. This technique extracts the subgraph of PSI from the malicious code, and the accuracy of this detection is around 97%, but the limitation over here is to seize the malicious samples. The evaluations of the survey are shown in the Table. 3

Very few papers have been published based on the windows based botnet. This survey aims to know the behavior of the emotet botnet, which will be useful in designing the botnet prevention techniques for windows architecture.

Table 3. Comparison of another survey

Survey	Published Time	Prevention Domain	Prevention Method
Tuan [15]	2022	DNS	(i) LA_Bin07 (ii) LA_Mulo7
Abrantes [16]	2022	Universal	(i) CICFlowMeter (ii) ML algorithm
Feizi [17]	2022	Universal	(i) Traffic behavior analysis
Al-Nawasrah[18]	2022	DNS	(i) Hybrid Supervised Fast-Flux Killer System (ADeSNN)
S.Kumar [19]	2020	IoT	(i) ML statistical analysis
Trajanovski[20]	2022	IoT	(i) TF-IDF (ii) DBSCAN
Yamaguchi [21]	2022	IoT	(i) Botnet Defense System (ii) White-Hat Worm
Al-Sarem [22]	2022	IoT	(i) Mutual Information Based Feature Selection with ML
Alphonse [23]	2022	IoT	(i) ML statistical analysis
Shinan [24]	2022	SDN	(i) ML statistical analysis
Hosseini [25]	2022	Universal	(i) Negative selection algorithm (ii) Convolution neural network (iii)Classification methods
Jithu [26]	2022	IoT	(i) Deep Neural Network (DNN)
Salim [27]	2021	lightweight container-based botnet	(i) LSTM algorithm
Mihajlović [28]	2021	Mobile	(ii) Convolutional neural networks (CNNs)
Raj [29]	2021	IoT	(i) one-class SVM (ML-based approach)
Soleymani [30]	2021	DNS	(i) Data pre-processing techniques in ML
Manzoor [31]	2021	Universal	(ii) ML statistical analysis
Rahmantyo [32]	2021	IoT	(i) Deep Residual 1DCNN
Aruna [33]	2021	IoT	(i) Artificial Intelligence
Sharmila [34]	2021	IoT	(i) KPCA Reduction Techniques
Joshi [35]	2020	Universal	(i) Fuzzy Logic-based feature engineering approach
Das [36]	2020	IoT	(i) lightweight cryptography Model
Fejrskov [37]	2020	DNS	(i) IP Generation Algorithm (IGA)
Ibrahim [38]	2020	IoT	(i) IoT- Susceptible-Infectious-Abandon (IoT-SIA) model
Amina [39]	2019	Windows-based botnet	(i) Bibliometric Analysis (No specific algorithm has been mentioned for prevention)
Soltani [40]	2014	Windows-based botnet	(ii) Detecting fast-flux service networks
Sinha [41]	2010	Mariposa botnet For windows	(i) Reverse Engineering Techniques (No specific algorithm has been mentioned for prevention)

3. Proposed Snort Based Botnet Detection System

3.1. Botnet

The botnet has grabbed the attention of cybersecurity professionals in COVID-19. In the last two years, a botnet has targeted many banking and finance sectors to perform the Distributed Denial of Services (DDoS) attack. A botnet is a network of compromised host devices used to perform malicious activities [5]. The computer system, IoT devices, smartphones, etc., are an example of host devices. Botnet typically consists of a bot server (C&C) and one or more bot clients. The botherder communicates with bot clients using an Internet Relay Chat (IRC) channel in the botnet network. Botnet performs the following steps. Step 1: The new bot client joins an IRC channel and listens to the commands. In Step 2: Each client retrieves the message sent by the botherder to the IRC server. Step 3: bot client retrieves the commands via the IRC channel. In Step 4, the bot client performs the task given by the server. In the final step, the bot client reports the outcome of the execution of the commands. The botnet life cycle is shown in Figure.2.

Botnet detection has become the highest priority as attackers are coming up with new malware attributes and pattern which is difficult to recognize by Antimalware,

Intrusion Detection System (IDS), Antivirus, Firewall, etc. The researcher introduced various detection and prevention mechanisms against the botnet on IoT, Software Defined Network (SDN), and Mobile. They have used various Machine Learning (ML), Deep Learning concepts, traffic behavior, etc. Very few techniques are introduced for windows botnet.

In this proposed work, the author has analyzed the latest live botnet: emotet, which is mainly responsible for launching a Trojan or DDoS attack on the windows platform. Emotet is one of the most dangerous botnets. In 2014, Trend Micros discovered Emotet as TrojanSpy.Win32.EMOTET.THIBEAI [46]. This Trojan was famous as a banking Trojan as it could steal the data by sniffing out the network activity. After that, every year, an emotet botnet has been encountered. On Nov 14, 2021, emotet returned to the threat landscape [47]. The malware distribution was done via the TrickBot malware and email campaigns. In this work, the author has created a dataset of IoCs based on the analysis. These IoCs will help create relevant snort rules to prevent the windows system from being against the emotet. Snort is open source instruction detection and prevention system. It supports the rules that help recognize the malicious activity in the network. Snort match packet with predefined rules and generate the alert message accordingly. The proposed model is designed in Figure 3

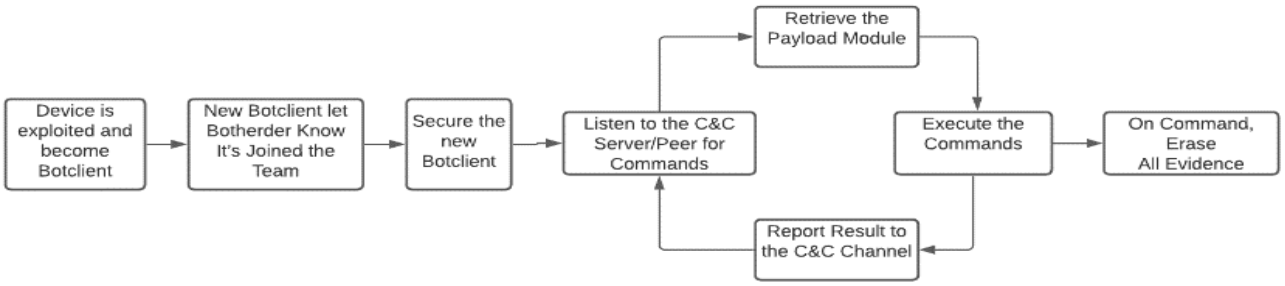


Fig. 2 Botnet Life Cycle

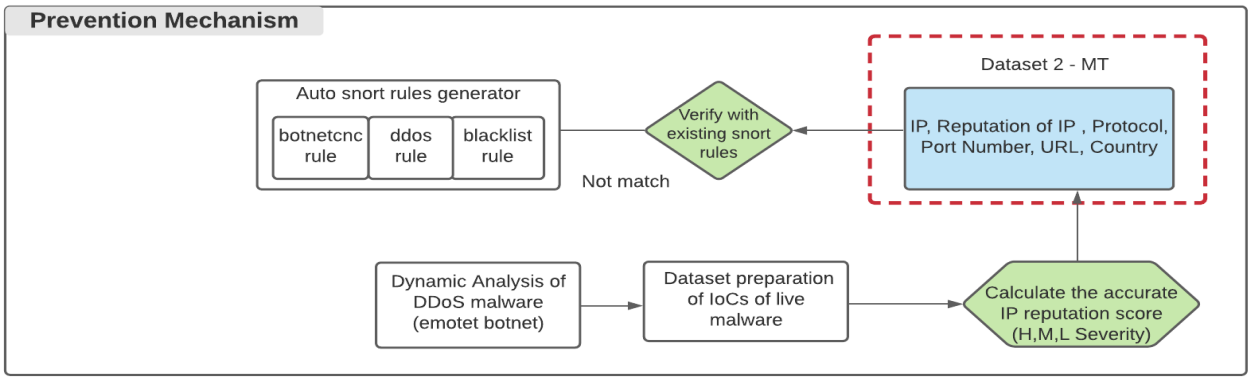
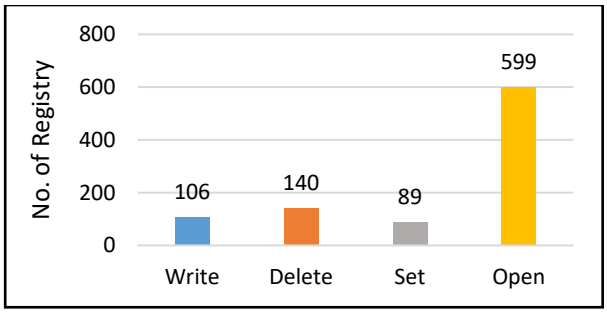


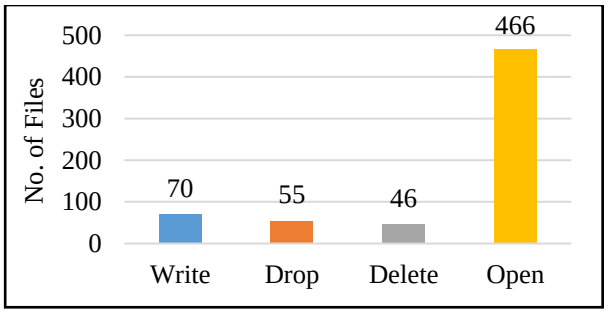
Fig 3. Snort based botnet detection mechanism

3.2. Dataset

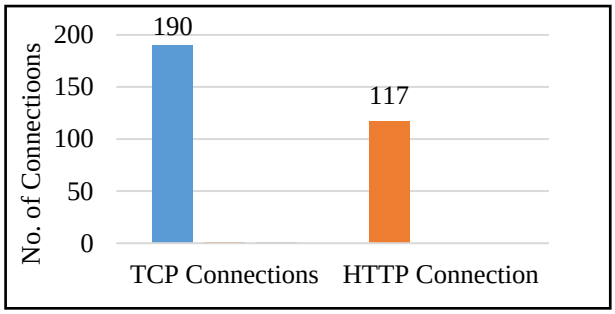
Malicious third parties have targeted the large windows systems. To overcome such issues, there is a need to effectively develop realistic protection and investigation countermeasures, such as network intrusion detection and forensic network systems. So to fulfill this purpose, a well-organized dataset is supreme for the training of the systems. In this paper, static and dynamic analysis of a live sample of emotet has been done. This dataset includes the features such as TCP connections, Registry operations, File operations, hash values, etc. Also, it includes actual botnet trafficking of 35 live samples. The samples have been analyzed using Pestudio, Virus Total, Alien Vault, and Gidhra static analysis. The malware sample has been run on IDA, Tencent HABO, Cuckoo sandbox, and Any for dynamic analysis. Run, Rising MOVES sandbox, and Hybrid Analysis SandBox. The final dataset of IoCs has been prepared after performing data cleaning and pre-processing. This dataset will be helpful to various researchers to understand the features of emotet. To highlight the real impact of botnet attacks involves a realistic network and traffic. In figure 4, the IoCs of emotet are shown. The results of IoCs are collected after analyzing the samples dynamically in 2 minutes. The final features after analyzing the emotet are listed in Table 4.



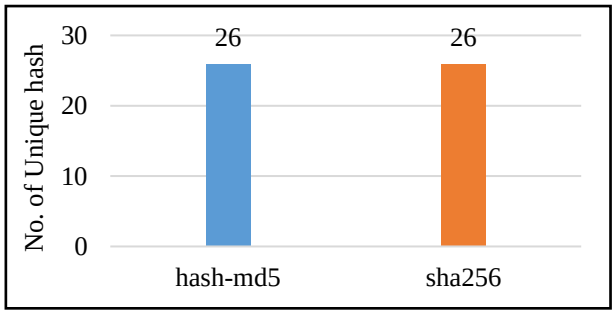
(a) Registry Operations



(b) File Operations



(c) Unique TCP & HTTP connection



(d) Unique hash-md5 & sha256

Fig 4. IoCs collected after analyzing emotet botnet

Table 4. emotet features

Sr.No	Feature	Description
1	NoReWr	Number of Registries written
2	NoReDe	Number of Registry Deleted
3	NoReSe	Number of Registry Set
4	NoReOp	Number of Registry Open
5	NoFiWr	Number of Files written
6	NoFiDr	Number of Files Dropped
7	NoFiDe	Number of Files Deleted
8	NoFiOp	Number of Files Open
9	Src_IP	Source IP address where connection has been established
10	Src_port	Source Port where connection has been established
11	Country	Country details where connection has been established
13	Protocol	Protocol details
14	Uni_hash_md5	Unique md5 hash value for each operation
15	Uni_Sha256	Unique sha256 hash value for each operation

4. Proposed Model

The proposed model is a signature-based model which involves the dataset of IP address, port, and country location. The proposed model calculates the IP reputation and differentiates it based on a severity score. This final IP, protocol, and port dataset will feed into the snort generator module. Snort follows the specific syntax to generate the rules shown in Table 5.

Table 5. Snort rules format

Rule Header							Rule Option
Action	Protocol	Source Address	Source Port	Direction	Destination Address	Destination Port	
alert	tcp	any	any	->	any	any	(msg:"TCP Packet is detected"; sid:1000010)

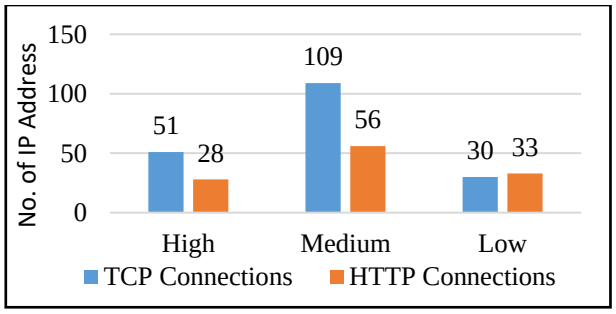
The signature must be uploaded to the rule file to check the generated signatures are valid. Here default signature is used with no additional parameter. Auto-generator generates the alerts based on the input value. If the severity is high,

snort will generate the alert message and block the traffic. If the severity is low, it will generate the message and allow the traffic based on the protocol and port number. The severity has been set based on the hypothetical value. Figure 5. (a) Shows the severity count of TCP and HTTP connections collected during the malware analysis.

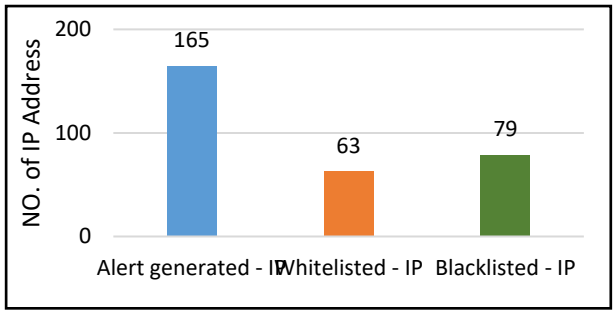
The Algorithm design for IP reputation is given below.

Input: emotetIPPortlist Output: Botnetcnc.rule,DDoS.rule, Blacklisted.rule
for each malwaresample S do Where S is 1 to 35
collect IoC ← Static + dynamic Analysis
emotetIPPortlist← emotet IP + emotet Port
for each emotetIPPortlist do
check IP with IP reputation engine
Calculate IP score Thresholds value r
if ["$r" = "$VAR1"] then
echo "$line, None"
elif [$r -lt 4] && [$r -gt 0] then
echo "$line, Medium"
elif [$r -gt 3] then
echo "$line, High"
elif ["$r" = "0"] then
echo "$line, Safe"
else echo "$line, Error"
endfor
endfor
for each traffic T # Verify malign traffic (MT) with existing botnet detection snort rule (EBDSR)
If ["emotetIPPortlist" = "EBDSR"] then
echo "no need to generate new rules"
else generate newrules.rule [botnetcnc, ddos, blacklisted]
endfor

Out of 190 TCP connections, 51 IPs have high severity, 109 IPs have medium severity, and 30 are safe. Same for HTTP connection, around 28 IPs are highly severe, 56 IPs have medium severity, and 33 IPs are safe. The number of rules generated by snort IDPS for emotet is shown in Figure 5 (b).



(a) Result of snort IDPS based no of IP address



(b) The severity of IP address

Fig 5. Result of snort IDPS based no IP signature

As shown in figure 6, out of a total of 307 IP addresses, 79 IP addresses will be blocked by snort IPS. For 165 IP addresses, snort will generate an alert message, and the request from 63 IP addresses will be transferred by snort in the network.

The dataset results show that the US keeps first place in a number of DDoS attacks. Figure 6 shows the No of unique TCP connections created by emotet per country. Germany has stepped out up to the second position (8.64%). Whereas Argentina shares (6.8%), landing the third country. The fourth place is held by France (6.36%), which is nearly the same as Argentina.

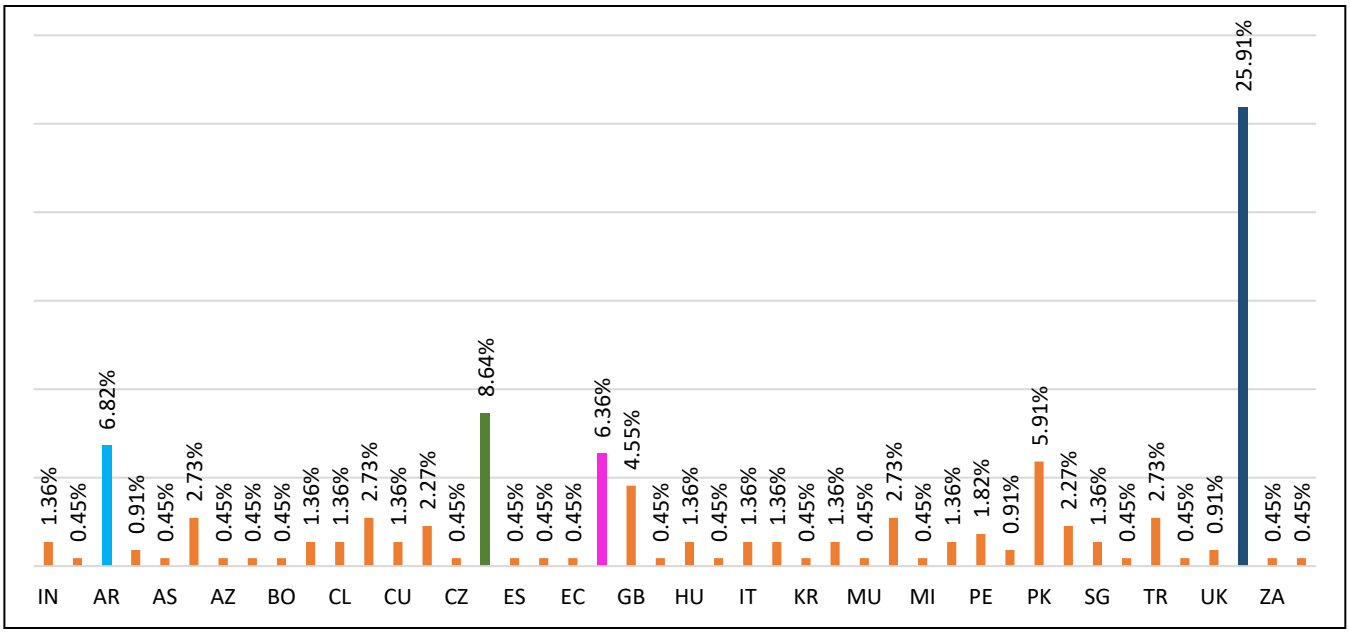


Fig 6. No unique TCP connections are created by emotet per country

5. Conclusion

In this work, the existing dataset has been analyzed, and presented a new dataset of emotet botnet. This dataset was developed on the bases of the live 35 emotet samples. The label features the attack flow, attack traffic, compromised system files, and registry details. As per the result analysis, first place has been held by the US (25.91%) for creating a connection with emotet. As per the observation, 79 malicious IP details are based on IP reputation. A total of 307 snorts rules have been generated to detect the emotet at an early stage. The additional features, e.g., DNS features, threat, output strings, and process information, will be considered in future work to generate more accurate results by IDS.

References

[1] (2020). The Prnewswire Website. [Online]. Available: https://www.prnewswire.com/news-releases/top-cyber-security experts-report-4-000-cyberattacks-a-day-since-covid-19-pandemic-301110157.html
[2] (2020). The Economic Times. [Online]. Available: https://economictimes.indiatimes.com/tech/internet/fearware-in-the-times-of-covid-19-pandemic/articleshow/75664689.cms?from=mdr
[3] (2022). The ZDNet Website. [Online]. Available: https://www.zdnet.com/article/2021-was-a-terrible-year-for-cybersecurity-without-action-2022-could-be-even-worse/
[4] Lee Wenke, Malware and Attack Technologies Knowledge Area Issue. (2020).
[5] Koroniotis, Nickolaos, et al., Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-Iot Dataset, Future Generation Computer Systems. 100 (2019) 779-796.
[6] Nkongolo, Mike, Jacobus Philippus van Deventer, and Sydney Mambwe Kasongo, UGRansome1819: A Novel Dataset for Anomaly Detection and Zero-Day Threats, Information. 12(10) (2021) 405.
[7] Özgür, Atilla, and Hamit Erdem, A Review of KDD99 Dataset Usage in Intrusion Detection and Machine Learning Between 2010 and 2015. (2016).
[8] I. Ali, A. I. A. Ahmed, A. Almogren, et al., Systematic Literature Review on Iot-Based Botnet Attack, IEEE Access. 8 (2020) 212220–212232.
[9] M. Singh, M. Singh, and S. Kaur, Issues and Challenges in DNS Based Botnet Detection: A Survey, Computers & Security. 86 (2019) 28–52.
[10] M. Sandip Sonawane, A Survey of Botnet and Botnet Detection Methods, International Journal of Engineering Research & Technology (IJERT). 7(12) (2018).
[11] K. Alieyan, A. Almomani, A. Manasrah, and M. M. Kadhum, A Survey of Botnet Detection Based on DNS, Neural Computing and Applications. 28(7) (2017) 1541–1558.
[12] X. Li, J. Wang, and X. Zhang, Botnet Detection Technology Based on DNS, Future Internet. 9(4) (2017) 55.
[13] Almutairi, Suzan, et al., Hybrid Botnet Detection Based on Host and Network Analysis, Journal of Computer Networks and Communications. (2020).
[14] Xing, Ying, et al., Survey on Botnet Detection Techniques: Classification, Methods, and Evaluation, Mathematical Problems in Engineering. (2021).
[15] Tuan, Tong Anh, Hoang Viet Long, and David Taniar, On Detecting and Classifying DGA Botnets and their Families, Computers & Security. 113 (2022) 102549.
[16] Abrantes, Rodrigo, Pedro Mestre, and António Cunha, Exploring Dataset Manipulation via Machine Learning for Botnet Traffic, Procedia Computer Science. 196 (2022) 133-141.
[17] Feizi, Sanaz, and Hamidreza Ghaffari, Detecting Botnet Using Traffic Behaviour Analysis and Extraction of Effective Flow Features, Intern
[18] Al-Nawasrah, Ahmad, et al., Botnet Attack Detection Using A Hybrid Supervised Fast-Flux Killer System, Journal of Web Engineering. (2022) 179-202.
[19] S. Kumar and B. R. Chandavarkar, DDOS Prevention in IoT, 12th International Conference on Computing Communication and Networking Technologies (ICCCNT). (2021) 1-6. Doi: 10.1109/ICCCNT51525.2021.9579765.

[20] Trajanovski, Tolijan, and Ning Zhang, An Automated Behaviour-Based Clustering of IoT Botnets, Future Internet. 14(1) (2022): 6.
[21] Yamaguchi, Shingo, and Brij Gupta. Botnet Defense System and White-Hat Worm Launch Strategy in IoT Network, Advances in Malware and Data-Driven Network Security, IGI Global. (2022) 127-147.
[22] Al-Sarem, Mohammed, et al., An Aggregated Mutual Information Based Feature Selection with Machine Learning Methods for Enhancing IoT Botnet Attack Detection, Sensors. 22(1) (2022) 185.
[23] Alphonse A, Sherly, EL Dhivya Priya, and M. Kowsigan, Review of Machine Learning Techniques Used for Intrusion and Malware Detection in WSNs and IoT Devices, Design and Development of Efficient Energy Systems. (2022) 57-65.
[24] Shinan, Khlood, et al., Machine Learning-Based Botnet Detection in Software-Defined Network: A Systematic Review, Symmetry. 13(5) (2021) 866.
[25] Hosseini, Soodeh, Ali Emamali Nezhad, and Hossein Seilani, Botnet Detection Using Negative Selection Algorithm, Convolution Neural Network and Classification Methods, Evolving Systems. (2021) 1-15.
[26] Jithu P. et al., Intrusion Detection System for IOT Botnet Attacks Using Deep Learning, SN Computer Science. 2(3) (2021) 1-8.
[27] Salim, Mikail Mohammed, Sushil Kumar Singh, and Jong Hyuk Park, Securing Smart Cities using LSTM Algorithm and Lightweight Containers Against Botnet Attacks, Applied Soft Computing. 113 (2021) 107859.
[28] Mihajlović S, D. Ivetić, and I. Berković, Use of CNNs on Mobile Devices to Protect Data from Malware and Unauthorized Attacks.
[29] Raj, Mehedi Hasan, et al., IoT Botnet Detection Using Various One-Class Classifiers, Vietnam. J. Comput. Sci. 8(2) (2021) 291-310.
[30] Soleymani, Ali, and Fatemeh Arabgol, A Novel Approach for Detecting DGA-Based Botnets in DNS Queries Using Machine Learning Techniques, Journal of Computer Networks and Communications. (2021).
[31] Manzoor, Nosheen, et al., Role of Machine Learning Techniques in Digital Forensic Investigation of Botnet Attacks, International Journal of Management (IJM). 12(2) (2021).
[32] Rahmantyo, D. Tsany, Bayu Erfianto, and G. Bayu Satrya, Deep Residual CNN for Preventing Botnet Attacks on the Internet of Things, 4th International Conference of Computer and Informatics Engineering (IC2IE), IEEE. (2021).
[33] Aruna J, and S. Prayla Shyry, Survey on Artificial Intelligence Based Resilient Recovery of Botnet Attack, 5th International Conference on Trends in Electronics and Informatics (ICOEI), IEEE. (2021).
[34] Sharmila B. S, and Rohini Nagapadma, Multi Core DNN based IDS for Botnet Attacks using KPCA Reduction Techniques. (2021).
[35] Joshi, Chirag, Ranjeet Kumar Ranjan, and Vishal Bharti, A Fuzzy Logic Based Feature Engineering Approach for Botnet Detection Using ANN, Journal of King Saud University-Computer and Information Sciences. (2021).
[36] Das, Suchitra, P. P. Amritha, and K. Praveen, Detection and Prevention of Mirai Attack, Soft Computing and Signal Processing, Springer, Singapore. (2021) 79-88.
[37] Fejrskov, Martin, et al., An Uneven Game of Hide and Seek: Hiding Botnet CnC by Encrypting IPs in DNS Records.
[38] Ibrahim, Mohammed, et al., The Impact of Memory-Efficient Bots on IoT-WSN Botnet Propagation, Wireless Personal Communications. (2021) 1-13.
[39] Amina, Shehu, et al., A Bibliometric Analysis of Botnet Detection Techniques, Handbook of Big Data and IoT Security. Springer, Cham. (2019) 345-365.
[40] Soltani, Somayeh, et al., A Survey on Real World Botnets and Detection Mechanisms, International Journal of Information and Network Security. 3(2) (2014):116.
[41] Sinha, Prosenjit, et al., Insights from the Analysis of the Mariposa Botnet, Fifth International Conference on Risks and Security of Internet and Systems (CRiSIS), IEEE. (2010).
[42] Kumar, Sunil, Bhanu Pratap Singh, and Vinesh Kumar, A Semantic Machine Learning Algorithm for Cyber Threat Detection and Monitoring Security, 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), IEEE. (2021).
[43] Evans Mwasiaji, Kenneth Iloka, Cyber Security Concerns and Competitiveness for Selected Medium Scale Manufacturing Enterprises in the Context of Covid-19 Pandemic in Kenya, SSRG International Journal of Computer Science and Engineering. 8(8) (2021) 1-7.
[44] Mahesh M. Baradkar, Dr.Bandu B. Meshram, A Survey on Cloud Security: Infrastructure as a Service, SSRG International Journal of Computer Science and Engineering. 6(6) (2019) 17-21.
[45] Li, Andrea, Privacy, Security and Trust Issues in Cloud Computing, International Journal of Computer Science Engineering. 6(10) (2019) 29-32.
[46] (2020). Emotet Now Spreads via Wi-Fi, Emotet Now Spreads via Wi-Fi. [Online]. Available: https://www.trendmicro.com/vinfo/de/security/news/cybercrime-and-digital-threats/emotet-now-spreads-via-wi-fi
[47] (2021). Schwarz D, & Kumar A, Return of Emotet: Malware Analysis, Zscaler. [Online]. Available: https://www.zscaler.com/blogs/security-research/return-emotet-malware-analysis
[48] Arshiya Moin, Artificial Intelligence Vs Covid19, SSRG International Journal of Computer Science and Engineering. 7(5) (2020) 5-7.