This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Malware Threat on Edge/Fog Computing Environments from Internet of Things Devices Perspective

IBRAHIM GULATAS[1], H. HAKAN KILINC[2], A. HALIM ZAIM[3] AND M. ALI AYDIN.[4]

[1]Department of Computer Engineering, Science Institute of Istanbul Commerce University, ISTANBUL-TURKEY (e-mail: ibrahim.gulatas@istanbulticaret.edu.tr)
[2]Department of Research and Development, Orion Innovation Turkey, ISTANBUL-TURKEY (e-mail: hakan.kilinc@orioninc.com)
[3]Department of Computer Engineering, Science Institute of Istanbul Commerce University, ISTANBUL-TURKEY (e-mail: azaim@istanbulticaret.edu.tr)
[4]Department of Computer Engineering, Engineering Faculty of Istanbul University-Cerrahpasa, ISTANBUL-TURKEY(e-mail: aydinali@istanbul.edu.tr)

Corresponding author: Ibrahim Gulatas (e-mail: ibrahim.gulatas@istanbulticaret.edu.tr).

ABSTRACT Developing a secured information processing environment highly depends on securing all of the layers and devices in the environment. Edge/Fog computing environments are not an exception in this case, and the security of those environments highly depends on securing the Internet of Things (IoT) devices which are the most vulnerable devices through the environment. The adoption of Edge/Fog computing paradigms by new emerging technologies stimulated malware development for IoT platforms. Recent attacks initiated by IoT malware show that these attacks make a tremendous impact on compromised systems in terms of Quality of Service because of the number of infected IoT devices. In the light of these developments, there is an enormous need for efficient solutions. However, defense capability against these new malware types is highly constrained by the limited understanding of these new emerging paradigms and the lack of access to malware samples. This paper mainly focuses on IoT malware to understand the behaviors of malware on the most vulnerable layer of the Edge/Fog computing environments. Mainly, 64 IoT malware families are located from 2008 when the first known IoT malware emerged to October 2022. These malware families are systematically characterized from various aspects, including target architecture, target device, delivery methods, attack vectors, persistence techniques, and their evolution from existing malware. During this characterization process, two different investigation frameworks, "Cyber Kill Chain" and "Mitre ATT&CK for ICS," have been adopted in different investigation layers. This paper aims to bring light to future researches with presented features of the IoT malware.

INDEX TERMS Edge/Fog Computing, Internet of Things (IoT), Malware, Malware Analysis, Malware Phylogeny, Security

I. INTRODUCTION

THE number of connected devices is increasing with colossal velocity. It is estimated that Internet of Things(IoT) devices will account for 50 percent (14.7 billion) of all networked devices by 2023 [1]. This increase in ubiquitous devices also brings new problem areas in terms of bandwidth consumption and latency between these devices and cloud servers. Edge Computing and Fog Computing paradigms emerged as a solution to this bottleneck [2]. The heterogeneous and distributed structure of these paradigms reveals some security leaks. As it is seen in the previous newly emerging technologies, the devices inside these environments are becoming a new target for malicious activities.

Moreover, Edge/Fog computing environments inherit all security vulnerabilities of previous technologies, which enable these environments to evolve, such as wireless sensor networks (WSNs), distributed P2P (peer-to-peer) systems, and virtualization platforms. Besides, these computing environments take advantage of their layered structures to provide a solution to the bottlenecks in the internet infrastructure; however, each layer of these structures has novel vulnerabilities. In this regard, one of the leading security flows is the attacks initiated by malware. Because of the heterogeneous architecture of the Edge/Fog Computing Environment, these systems are a target for malware developed for various architectures.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

Edge devices are taking a big part with a huge number of areas of utilization in these new computing ecosystems. We believe that the constrained nature of edge devices makes them the weakest point in the security chain of the Edge/Fog Computing Environment. Only one exploit in one device may pose a threat to all devices in the same environment. Recent reports show that IoT devices are one of the most popular targets among malicious people in recent years. According to the Cyber Threat Report of Sonicwall company, 57 million IoT malware attacks have been detected in the first six months of 2022 which indicates a 77 percent increase from the first six months of the previous year [3]. This 6-months attack volume is higher than yearly attacks recorded in each of 2018, 2019, and 2020. The number of IoT malware families may be limited for now compared to malware that infects conventional devices. However, the vast increase in IoT devices will also lead to an increase in malicious software numbers.

The recent attacks initiated by infected IoT devices reveal the importance of the situation. The infamous Mirai attack is one of the most remarkable examples of IoT botnets [4]. IoT devices infected by Mirai malware are used in a botnet to initiate one of the biggest Distributed Denial of Service (DDoS) attacks in the history. The volume of the generated data traffic was a new record in information technology. During the peak of the attack, 600 Gbps data traffic is generated by the least powerful devices [5].

Unfortunately, Mirai is not the last example of IoT botnets. Quite the contrary, it was just the beginning of the era of IoT botnets. In this research, we had located 21 different malware families from 2008 when the first known IoT malware "Hydra" was firstly seen in the wild to the September 2016 "First Seen In The Wild" date of Mirai malware. On the other hand, 43 different malware families are located from Mirai to October 2022. Also, 29 of these 43 malware families are variants of Mirai malware, which indicates that these malware families inherited some features of Mirai malware. This huge increase in the IoT malware types is mainly caused by the leakage of the Mirai source code on the publicly available code repositories. For this reason, the IoT malware evolution process could be classified as "Before Mirai" and "After Mirai".

Even though there are some invaluable researches on the IoT Malware domain, none of them reflects the whole attack scope of IoT Malware. This study aims to reflect the state-of-the-art IoT malware threat by utilizing research papers, technical reports of antivirus (AV) vendors, and blog posts of malware researchers. In the scope of this research, research papers, technical reports of AV vendors, and blog posts of malware researchers related to IoT malware are collected for the period between 2008 and October 2022. During our research, we assessed that the efforts of AV vendors are far more informative than academia. AV vendors and some malware researchers published their analysis results in a timely manner to warn their subscribers about recent threats in the IoT malware domain. For this reason, this research gives summarized information on all those reports in a framework that will be described hereinafter.

The main motivation behind this research is to introduce IoT Malware Families along with their behaviors and features to bring light for further research which aims to contribute to the security of Edge/Fog Computing Environments by securing IoT Devices. The main contributions of this research are summarized in three stages.

- 64 different malware families which effects IoT devices are located. This comprehensive collection reflects the majority of IoT malware if it is not all. Besides, most of these malware families are becoming a topic in academia for the first time.
- A novel Phylogenetic Tree of IoT Malware has been presented in the scope of this research to gain an understanding of the evolvement of IoT Malware.
- Behavioral analysis of located malware families presented in a three-layered approach. With this research, The Cyber Kill Chain Framework and Mitre ATT&CK (Adversarial Tactics, Techniques, and Common Knowledge) for ICS (Industrial Control Systems) matrix is used collaboratively to present behaviors of IoT malware for the first time in the academia.

As herein-before stated, the main goals and contributions of this research can be classified into three stages. First, this research fulfills the need for presenting an extensive collection of IoT malware families. Within this research, 64 different malware families are located from 2008 to October 2022. We believe that this collection reflects the majority, if it is not all, of the known IoT malware families by the date of the published time of this paper. Most of these malware families are covered in an academic research for the first time. Botnet attacks are the most common attack type among the IoT malware families. Infected IoT devices are used in a botnet to initiate DDoS attacks. However, the attack types are not only limited to create botnets but also mining cryptocurrencies, and DNS Poisoning attacks are in scope.

This research's second goal and contribution is to reveal the evolutionary progress of the IoT malware families. As a state-of-the-art, Phylogenetic Tree of IoT Malware is built up. Four different malware families Hydra, Tsunami, Gafgyt, and Mirai, are determined as parent malware families and highly inherited by other malware families.

Last but not least, the third goal and contribution of this research is presenting a behavioral analysis of the detected malware families. We analyzed the behaviors of the IoT malware in a three-layered approach. The Cyber Kill Chain framework is adopted as a high-level investigation framework. Mitre ATT&CK framework which is becoming an industry standard knowledge base for presenting adversarial tactics and techniques for different platforms, was adopted as a mid-level framework. Mitre framework is not only becoming an industry standard but also academia started to apply this framework to researches. Gittins and Soltys utilize Mitre framework to present only Persistence tactics and techniques used by malware [6]. Kwon et al. developed a tool called

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

as Cyber Threat Dictionary which offers solutions for the threats by mapping Mitre ATT&CK framework to the NIST Cyber Security Framework [7]. With our research Mitre Framework is used for presenting the behavioral analysis of IoT malware for the first time in the academia. Also, this research presents the most comprehensive utilisation of Mitre ATT&CK framework for IoT malware with covering all of the attack phases of the Cyber Kill Chain Framework. Lastly, as a low-level investigation, attack vectors, communication ports, and protocols are presented. The low-level behavioral analysis shows that brute force attacks are the most common attack vector for gaining initial access to the victim devices. The most common attack vectors to compromise target devices are flooding attacks used for DDoS. Except for the five P2P communicating malware, all the rest of the malware communicates with a C2 (Command and Control) server to get the attack commands.

The rest of this survey is organized as follows. The next section is about previously conducted studies by other researchers. The Methodology of this research is explained in Section III. Malware types affecting IoT devices are introduced in Section IV. The evolution of IoT malware types is summarized in Section V. Behavioral Analysis of Detected Malware Families is given in Section VI. Finally, in Section VII, open issues are highlighted, and new directions for the research community to help them develop more precise detection mechanisms have been proposed.

II. RELATED WORK

The previous studies conducted in the IoT malware domain are presented in three subcategories as surveys and general informative research papers, IoT malware datasets, and IoT malware analysis and detection researches.

A. IOT MALWARE SURVEYS

Wang et al. categorized IoT malware only according to infection techniques and analyzed only three families as an example [8]. Vignau et al. present features of the most commonly seen IoT malware families in the first 10 years of the IoT Malware History and they present a Phylogenic Graph which contains 16 malware families [9]. In this research, they focus on attack vectors and feature propagation between the malware families. Two years later, Vignau et al. presented another survey research that investigates IoT malware between 2008 to 2019 [10]. They again presented feature propagation between the malware families but this time their research was not only limited to attack vectors, they also added other features of the malware such as infection method, C2 communication, persistence methods, etc. Their research contains 28 different IoT malware families. Even though, they state that their malware collection reflects all active botnets between 2008 and 2019, they only covered 28 malware families, while we are able to locate 42 different malware families for the same period. Alrawi et al. compared IoT malware with desktop and mobile malware. They presented general insight about some features of IoT malware such as infection methods, attack vectors, and persistence techniques, but their work didn't contain specifically adopted techniques by each malware family [11].

During our research on previously conducted survey studies in IoT Malware, we detect that these studies give the general perspective of IoT malware behaviors. These studies cover only a limited number of malware families. Besides, they present only a few characteristics of the malware families, and there is no behavioral information about most of the IoT malware families. In our study, we aim to present detailed information on our malware collection, and to the best of our knowledge, our study covers all of the IoT malware families since 2008. A comparison of the survey studies in IoT malware is given in Table I.

B. IOT MALWARE DATASETS

The rampant growth of IoT malware, as hereinbefore stated, shows that effective mitigation methods or defense mechanisms are highly needed. The key to protecting systems effectively highly depends on understanding attacks. However, the lack of a comprehensive IoT malware dataset makes it almost impossible for researchers to understand IoT malware attacks. There are only a few publicly available IoT malware datasets. One of the publicly available datasets created for IoT malware was presented by Pa et al. They deployed a honeypot called IoTPot to collect active IoT malware [12]. They have published their findings and collected malware binaries with a dataset that contains 86.496 malware binary files. Another dataset was presented by Azmoodeh et al. that contains only 128 malware samples of ARM-based IoT applications between February 2015, and January 2017 [13]. Also, capturing network traffic that belongs to compromised IoT devices is an applicable method for creating datasets. Meidan et al. presented the N-BaIoT dataset [14]. Their dataset was created by injecting Mirai and Bashlite malware into 9 commercial IoT devices which are very few to present the general of the IoT malware cluster. The dataset contains 7062606 instances of network traffic data. Garcia et al. presented the IoT-23 dataset that includes network traffic from 20 malware families, which are again very few to present the general of the IoT malware cluster [15]. Safaei Pour et al also present a highly comprehensive dataset that contains 3.6TB of network traffic created by 440.000 compromised IoT devices [16]. Trajanovski and Ning presented one of the latest and most comprehensive IoT malware dataset [17]. They proposed an IoT Botnet Detection and Analysis (IoT-BDA) framework which combines a honeypot and a sandbox. During 7 months period, their framework collected 4077 malware samples and they presented the analysis results with a dataset that covers static analysis features, dynamic analysis (System Calls and Network Traffic Capture), and malware sample files (ELF Binaries). Nevertheless, their static analysis only covers binary analysis results such as symbols and linking information but lacks OPCODE sequences which is the main output of a static analysis. Another popular technique among researchers is creating their own datasets for their research.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

TABLE 1: Comparison of IoT Malware Surveys

Research	Covered Number of Malware Families	Malware Lineage	Target Architecture	Target Devices	Infection Techniques	Exploited Vulnerabilities	C2 Communications	Attack Vectors	Persistance and Anti-analysis Techniques
Wang et al. (2017)	3	-	-	-	+	-	-	-	-
Vignau et al. (2019)	16	+	-	-	-	-	-	+	-
Vignau et al. (2021)	28	-	+	-	+	+	+	+	+
Alrawi et al. (2021)	16	-	-	-	+	+	+	+	+
This Research	64	+	+	+	+	+	+	+	+

This method is highly effective in evaluating the proposed methods; however, the lack of a commonly adopted dataset prevents comparing the evaluation metrics of the proposed methods. Botacin et al. presented challenges and pitfalls in the malware research area and highlight the usage of non-publicly available datasets limits reproducibility and prevents comparison of evaluation metrics [18]. The comparison of IoT Malware datasets is given in Table II.

TABLE 2: Comparison of IoT Malware Datasets

Research	Records	Binary Files	OPCODE	Symbols	Network Traffic	System Calls
Pa et al.- IoTPoT (2015)	86.496	+	-	-	-	-
Azmoodeh et al. IoT Dataset (2018)	128	-	+	-	-	-
Meidan et al - N-BaIoT (2018)	7.062.606	-	-	-	+	-
Garcia et al.- IoT-23 (2020)	23	-	-	-	+	-
Safai Pour et al. (2020)	440.000	-	-	-	+	-
Trajanovski and Ning (2021)	4.077	+	-	+	+	+

C. IOT MALWARE DETECTION

IoT malware detection is a relatively new research area compared to classical PCs and mobile devices. However, as it is mentioned in the previous sections, the impact of IoT malware should not be regarded as little, quite the contrary, the Mirai example showed that it can be more dangerous than classical malware. Even though there is still a long way to go to secure IoT devices against malware attacks, there are successful efforts for detecting IoT malware. Most of the IoT malware detection research is based on static analysis. Haddadpajouh et al. used their own OPCODE based dataset called as IoT Dataset with Recurrent Neural Network (RNN) and achieved an accuracy rate of 98,18% [19]. Su et al used the IoTPoT dataset and converted the malware binary files into gray-scale images [20]. They used Convolutional Neural Network (CNN) algorithm for classification and achieved an accuracy rate of 94%. Alasmary et al. created their own dataset and compared the characteristics of IoT malware and Android malware which are also mainly based on Linux by utilizing Control Flow Graphs [21]. Also, they proposed a detection mechanism based on CNN which gives a very high accuracy rate of 99.66%. Dovom et al. used IoT Dataset and achieved an average accuracy rate of 96.41% with Fuzzy Pattern Tree algorithm [22]. Darabian et al. used the IoT Dataset with Decision Tree, KNN, Random-Forest, MLP, SVM, and AdaBoost algorithms and they achieved the highest accuracy rate of 99,80% with the Decision Tree classifier. [23]. Vasan et al. presented a Cross-Architecture IoT Malware Detection method called as MTHAEL [24]. They build their own dataset and their method is based on Neural Network Advanced Ensemble Learning combining RNN and CNN with a very high accuracy rate of 99.98%. Another research for securing the Edge Computing ecosystem against IoT malware is conducted by Haddadpajouh et al. [25]. They used their previous dataset called as the IoT Dataset and proposed a malware detection mechanism for a cloud-edge gateway layer based on Support Vector Machine (SVM) by utilizing Grey Wolf Optimization (GWO) for feature selection with a very high accuracy rate of 99.72%.

On the other hand, there are some efforts based on dynamic analysis, and most of them are based on the investigation of network traffic. Meidan et al. used their own dataset N-BaIoT with Artificial Neural Network (ANN) and achieved TPR rate of 100%, and FPR rate of 1% [14]. Jeon et al. conducted a dynamic analysis for malware detection based on Convolution Neural Network (CNN) [26]. They used their own dataset and convert the behavioral data of IoT malware into images to analyze and detect malware, and with this effort, they reach a high accuracy rate of 99.28%. Rey et al. used the N-BaIoT dataset to detect IoT Malware [27]. They applied Federated Learning (FL) for two different models as supervised and unsupervised models, for the supervised model they applied Multi-Layer Perceptron (MLP) which obtains an accuracy rate of 99,38%, and for the unsupervised model, they applied ANN which obtains a TPR rate of 99,98%, and TNR rate of 91,78%. The comparison of IoT Malware Detection Research is given in Table III.

III. METHODOLOGY

This study follows a systematic methodology to provide a comprehensive approach for revealing malware threats on the edge devices layer of edge/fog computing environments. In this regard, firstly to identify the names of IoT malware families, exhaustive research has been conducted on academic papers, malware databases such as VirusTotal and MalwareBazaar, malware threat alarms of AV vendors, and OSINT(Open Source Intelligence) sources such as social media platforms and forums. Secondly, malware analysis reports of the identified malware families are collected from academic papers, technical reports of AV vendors, and blog posts of malware researchers. Besides, the source code of some of these malware families are publicly available on the code repositories such as GitHub. These source codes are also added to our collection. As a result of this step, we collected 470 academic studies and industrial reports which were published between 2008 and October 2022. Lastly, these scientific papers and industrial reports are investigated manually. During this investigation, the scientific papers and industrial reports which does not contain information related

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

TABLE 3: Comparison of IoT Malware Detection Research

Research	Dataset	Analysis Technique	Feature	Algorithm	Performance
Haddadpajouh et al. (2018)	IoT Dataset	Static	OPCODE	RNN	Accuracy:98,18
Su et al. (2018)	IoTPoT	Static	Malware Binary	CNN	Accuracy:94
Meidan et al. (2018)	N-BaIoT	Dynamic	Network Traffic	ANN	TPR:100 FPR:0,01
Alasmary et al. (2019)	Own Dataset	Static	OPCODE	CNN	Accuracy:99,66
Dovom et al. (2019)	IoT Dataset	Static	OPCODE	Fuzzy Pattern Tree	Accuracy:96,41
Darabian et al. (2020)	IoT Dataset	Static	OPCODE	Decision Tree	Accuracy:99,80
Vasan et al. (2020)	Own Dataset	Static	OPCODE	RNN+CNN	Accuracy:99,98
Jeon et al. (2020)	Own Dataset	Dynamic	Network Traffic +System call	CNN	Accuracy:99,28
Haddadpajouh et al. (2021)	IoT Dataset	Static	OPCODE	SVM+GWO	Accuracy:99,72
Rey et al. (2022)	N-BaIoT	Dynamic	Network Traffic	FL (MLP and ANN)	MLP Accuracy: 99,38 ANN TPR:99,98 TNR:91,78

to the contribution of this research are excluded. Moreover, the industrial reports may raise an issue about the validity of the provided information. For this reason, reports which are only generated by reverse engineering efforts and supported by screenshots of the conducted steps are taken into consideration while the others are omitted from the collection. As a result of these efforts, our collection contains 91 records containing 8 source codes, 13 scientific papers, and 70 industrial reports. The general overview of our methodology is presented in Fig. 1.

The scientific papers, industrial reports, and source codes within the collection are reviewed to achieve the goals of this research which are defined in Section I. Firstly, 64 IoT malware families are located and investigated in terms of "First Detection Date", "Target Architecture", "Target Device" and "Attack Type". The "First Detection Date" is determined by comparing the "First Seen In The Wild" value of VirusTotal and the detection date mentioned in the relevant report. According to this comparison, the earliest date is assessed as the "First Detection Date" because sometimes the researchers may submit malware samples after their malware analysis process. The "Target Architecture" is determined by combining the findings of the relevant reports and the "ELF Header Information" section of the VirusTotal. The "Target Device" and "Attack Type" information are extracted from the findings of the relevant reports. As a result, general information about the malware families which affects IoT devices is presented in Section IV.

Secondly, the Phylogenetic Tree of IoT Malware has been generated to provide an understanding about the evolvement

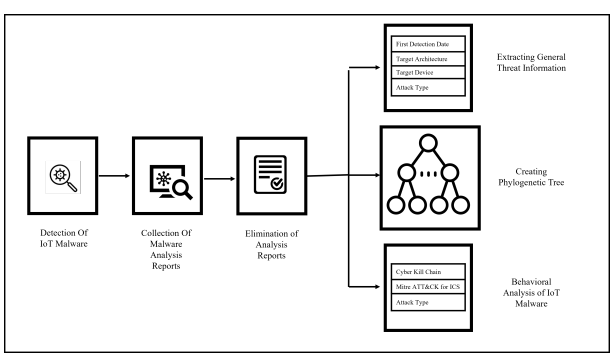


FIGURE 1: General Overview of Methodology.

of IoT Malware. Computer Virus Phylogenies was first introduced by Goldberg et al. in 1998 by inspiring biological species evolution. They define Malware Phylogenies as the "evolutionary history of computer viruses" [28]. Besides, Allix et al. showed that most of the malware families are generated by slight modifications in the source code of existing malware [29]. From this point of view, the inherited features between the malware families are collected from scientific papers, industrial reports, and source codes of the malware. The results of these efforts are presented in Section V.

Lastly, the third goal of this research is to illustrate the behavioral analysis of malware families by adopting an investigation framework from the high level to the low level. The ICS Cyber Kill Chain framework of SANS organization has been adopted for high-level investigation of IoT Malware [30]. This framework contains five main and seven subphases.

For the mid-level investigation, Mitre ATT&CK for ICS framework has been adopted [31]. This framework is a knowledge base that contains malicious TTP (tactics, techniques, and procedures). Mitre ATT&CK has three different investigation matrices for enterprise devices, mobile devices, and ICS systems. The enterprise and mobile device matrices are also divided into subcategories in terms of the target operating system. Unfortunately, there is no matrix for IoT devices and there are two different approaches for investigating IoT malware with Mitre's frameworks. Since all of the located malware families for IoT devices based on Linux, enterprise framework for Linux is adopted by some AV vendors. However, we believe that this framework is more applicable to enterprise devices. In this research, ATT&CK for ICS has been adopted because ICS devices are more similar to target devices of located malware families. ATT&CK for ICS framework has a comprehensive matrix to point out adversarial behaviors on the target device. This matrix consists of 11 tactics and 96 techniques. On the matrix, tactics represent the tactical objective of malicious activity while techniques represent the performed action to reach that tactical objection. For example, the adversaries' tactical objective is to manipulate, interrupt, or destroy the target system, defined as the "Impact" tactic in the ATT&CK matrix. To achieve this tactical objective, the attacker could benefit from one or more of the eleven techniques under this tactic.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

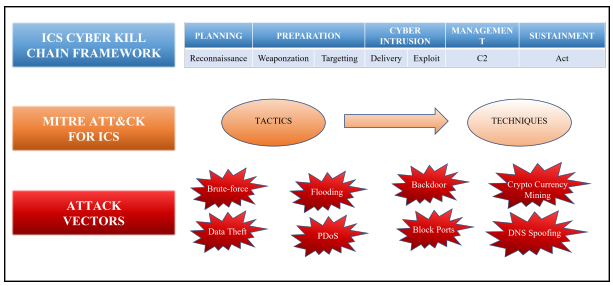


FIGURE 2: Cyber Kill Chain Framework.

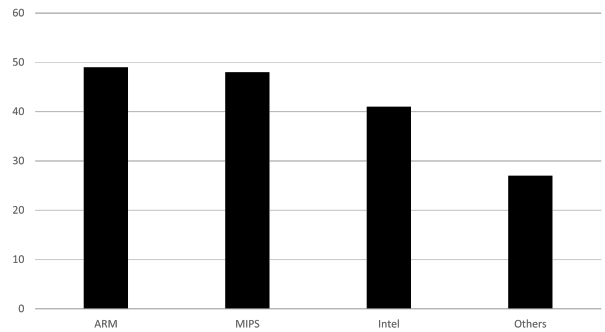


FIGURE 3: Target Architecture Distribution.

For low-level investigation attack vectors, exploited vulnerabilities, communication ports, and services are presented in a systematic manner. As a result of this three-layered investigation, a novel behavioral analysis framework is generated. The first layer covers the ICS Cyber Kill Chain Framework. In the second layer, the tactics and techniques of each phase in the ICS Cyber Kill Chain Framework are presented. In the third layer, attack vectors are highlighted as a part of the technique. The overview of our behavioral analysis framework is shown in Fig. 2 and the results of the behavioral analysis are presented in Section VI.

IV. MALWARE FAMILIES AFFECTING IOT DEVICES

The Hydra malware, the first known malware affecting IoT systems, started a new era for malicious code developers. After the source code of this malware was released publicly on the code repositories [32], new malware families evolved based on this source code. Since 2008, the number of IoT malware families increased with colossal velocity. The first goal of this research is to present the majority, if it is not all, of the IoT malware families to provide a deep understanding of the behaviors of these malware families for the research community to develop more precise defense mechanisms. As a result of our efforts, 64 different malware families are located. By the date of the published time of this research, we believe this collection reflects the state of the art of IoT malware. Brief information about these malware families is given in Table IV.

IoT malware is generally used for creating botnets to initiate DDoS attacks. The 51 (80%) malware families are used for creating botnets. However, the attack types are not only limited to creating botnets but also cryptocurrency mining, DNS Poisoning, Permanent Denial of Service (PDoS), and Data Exfiltration attacks are in scope. Besides, 12 (19%) of the malware use more than one attack type to compromise the target. Another dramatic statistic is that 13 (20%) of malware families (Carna, Linux.Darlloz, Linux.Wifatch, Brickerbot, PNScan, Moose, Linux.MulDrop, VPNFilter,
LiquorBot, Silex, Sora, ZuoRAT, and Shikitega) which represents 20 percent of our collection, are not used for DDoS attacks. Moreover, there are two White-Hat Trojans in this collection (Carna and Linux.Wifatch). These White-Hat Trojans were developed by security professionals to emphasize the security flows of IoT devices.

Most of the malware families target more than one CPU architecture, and they have different binary files for each of the target architectures. This diversity in the target creates a new challenge for signature-based detection approaches. Because the binary executables of the malware change according to the target CPU architecture, there are different signatures for the same malware family. Besides, most of the IoT malware detection methods are based on machine learning models. As it is discussed in Section II most of these efforts are based on OPCODEs which are obtained by static analysis of malware. Since IoT devices have diversity in terms of CPU architecture, the OPCODEs also vary according to the executed CPU architecture. This situation maintains a big challenge in building an effective malware detection method based on OPCODE. Developing an anti-malware tool based on machine learning techniques requires separate training processes for each of the target CPU architectures.

ARM and MIPS are the most common target architectures for IoT malware. There are 49 malware families which are targeting ARM, while this number is 48 for MIPS-based devices. Intel architectures are also on target with 41 malware families. PowerPC, Motorola, and Sparc architectures are susceptible to malware as well. Besides, some malware families like LuaBot and BrickerBot target only specific architectures to attack. The distribution of the target architecture is given in Fig. 3.

Network devices such as routers are leading target devices for IoT malware. The list of target devices is given in Table IV; however, we believe that the target devices are not limited to those devices. IP cameras, DVRs (Digital Video Recorders), toys, CCTV cameras, and TV receivers are infected by IoT malware as well. We assess that IoT malware can infect any device if they have relevant vulnerabilities.

V. EVOLUTION OF MALWARE FAMILIES

As it is mentioned hereinbefore, Hydra is the first known malware family for edge devices. It is not surprising to see the development of other malware families after the public release of the source code of the Hydra malware. After 2014, the rate of increase in IoT malware increased as IoT devices getting popular. After the infamous Mirai attack in 2016, the IoT malware threat takes the attention of all information

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

TABLE 4: Summary of IoT Malware

Malware Familiy	First Detection	Target Architecture	Target Device	Attack Type	Malware Familiy	First Detection	Target Architecture	Target Device	Attack Type
Hydra [33]	Feb2008	MIPSel, X86	D-link Routers	Botnet	**Fbot** [34], [35]	Sep 2018	ARM, MIPS, X86	HiSilicon DVR/NVR Soc	Botnet
Psyb0t [36], [37]	Jan 2009	MIPSel	Routers, Modems	Botnet	**Miori** [38]	Dec 2018	ARM, MIPS, X86 MIPSel, ARC, PPC, SH4	Routers	Botnet
Chuck Norris [36], [39]	Dec 2009	MIPSel	Modems, Routers	Botnet, DNS Spoofing, Device Config	**LiquorBot** [40]	May 2019	ARM, MIPS, X86 X64	Routers	Crypto Currency (Monero) Miner
Tsunami/ Kaiten [41], [42]	2010	MIPS	Zyxel Draytek Products	Botnet	**Silex** [43]	Jun 2019	ARM, MIPS, X86 SH4, Sparc	Any Device	Device Evasion
Aidra/Zendran Light Aidra/ [44], [45]	Jan 2012	ARM, MIPS, PPC MIPSEL, SH4, x86	Routers, Modems	Botnet	**Moobot** [46]–[48]	Jul 2019	AR, MIPS, X86	Routers, Modems DVRs, IP Cameras	Botnet
Carna [49], [50]	Apr 2012	MIPS	Any Device	White-Hat Trojan	**Echobot** [51], [52]	Aug 2019	ARM, MIPS, X86, m68K	Mitsubishi I-IoT Barracuda Firewall Citrix NetScale I-IoT	Botnet
Linux.Darlloz [53]	Nov 2013	ARM, MIPS, MIPSel, PPC	Webcams, Toys Set Top Boxes	Crypto Cur. Mining (Mincoin, Dogecoin)	**AirDropBot** [54]	Sep 2019	MIPS, M68k-68, Hnios2, Microblazebe, Microblazeel, SH4, Xtensa, ARC	Network Devices	Botnet
Linux.Wifatch [55], [56]	2014	ARM, MIPS, PPC SH4, x86	Any Device	White-Hat Trojan	**Ttint** [57]	Nov 2019	ARM, MIPS, X64, SH4	Tenda Routers	Botnet DNS Spoofing Data Exfiltration
TheMoon [58]	Feb 2014	ARM, MIPS	Linksys Routers	Botnet	**Dark Nexus** [59], [60]	Dec 2019	ARM, MIPS, MIPSel, X86, X64, Sparc, m68k, PPC, ARC, SH4	Any Device	Botnet
Setag/Mayday Elknot/BillGates [61], [62]	Feb 2014	X86, X64	Access Points, Servers	Botnet, Crypto Cur. Mining	**Mozi** [63], [64]	Dec 2019	ARM, MIPSel	Routers, DVRs	Botnet Data Exfiltration
Dofloo/Spike/ Mr.Black/Wrkatk/ Sotdas/DnsAmp AES.DDoS [65], [66]	May 2014	ARM, MIPS	Routers, Modems	Botnet	**Sora** [67]	Jan 2020	ARM, MIPS, MIPSel, X86, PPC, SH4, m68k SH4	Storage Systems Smart TVs, Routers	Device Evasion
Gafgyt/ Bashlite/ Lizkebab/ Torlus [68]–[71]	2014	ARM, MIPS, X86, PPC, SH4, m68k, Sparc	Routers, IP Cameras, DVRs	Botnet	**Rhombus** [72]	Feb 2020	X86, X64	Any Device	Botnet
XOR.DDoS [73], [74]	Sep 2014	X86	Any Device	Botnet	**Hoho** [75]	Feb 2020	ARM, MIPS, MISPSel, X86, PPC	Any Device	Botnet
ChinaZ/Znaich [76]	Jan 2015	X86	Routers, Servers, NAS	Botnet	**Mukashi** [77], [78]	Mar 2020	ARM, MIPS, MISPSel, X86, SH4	Zyxel NAS Devices	Botnet
Moose [79]	May 2015	ARM, MIPS	Routers, Modems	Social Media Hijacking, DNS Poisoning, MITM	**Vbot** [80]	Apr 2020	ARM, MIPS, MISPSel, X86, PPC	Any Device	Botnet
PNScan [81], [82]	Sep 2015	ARM, MIPS MIPSel, X86	Linksys Routers Fritz!Routers	MITM DNS Poisoning	**Kaiji** [83]	May 2020	ARM, X86, X64	Any Device	Botnet
Remaiten / KTN RM [84]	Mar 2016	ARM, MIPS MIPSel	Any Device	Botnet	**Katana** [85]	Oct 2020	ARM, MIPS X86, PPC	Routers	Botnet
Amesia [86]	Mar 2016	ARM, MIPS, PPC m68k, Sparc	DVRs	Botnet Device Evasion (wiping)	**HeH** [87]	Oct 2020	ARM, MIPS X86, PPC	Any Device	Botnet
NYA.Drop [88]	May 2016	MIPS	Routers, DVRs, CCTV Cameras	Botnet	**ZHtrap** [89], [90]	Feb 2021	ARM, MIPS, X86	Routers, Modems	Botnet
Radiation [91]	May 2016	ARM, MIPS, X86, MIPSel, PPC, X64, m68k, Sparc	CCTV Cameras	Botnet	**Unstable** [67]	Feb 2021	ARM, MIPS, X86	Routers, Modems	Botnet
LuaBot [92]	Aug 2016	ARM	Routers, Modems	Botnet	**Dark.IoT** [93], [94]	Feb 2021	ARM, MIPS, MISPSel, X86, SH4 M68K, PPC	Routers, Modems, NAS Devices	Botnet
Mirai [95]–[97]	Sep 2016	ARM, MIPS, X64	Routers, NAS IP Cameras Printer, DVRs TV Receivers	Botnet	**Sbidiot** [98], [99]	Mar 2021	ARM, MIPS MIPSel, X86	ZTE Routers	Botnet
Hajime [100], [101]	Oct 2016	ARM, MIPS MIPSel, X64	Routers, Modems, Gateways, DVRs, Printers	Botnet	**Scsihelper** [102]	Jul 2021	ARM, MIPS, X86	Any Device	Botnet
IRCTelnet/ New Aidra [103]	Oct 2016	ARM, MIPS, PPC	Routers, Modems DVRs, IP Cameras	Botnet	**Botenago** [104]	Nov 2021	MIPS, X64	Any Device	Botnet
Brickerbot [105]	Mar 2017	Busy Box OS	Webcams, Toys, Smart Bulbs	Device Evasion	**Manga/ Dark Mirai** [106]	Nov 2021	ARM, MIPS X86	TP Link Routers	Botnet
Persirai [107]	Apr 2017	ARM, MIPS, MISPSel, X86	IP Cameras	Botnet	**BeastMode** [108]	Mar 2022	ARM, MIPS, MISPSel, X86, SH4, X64, M68K, PPC	Routers, IP Cameras, NAS Devices,	Botnet
Linux.MulDrop [109]	May 2017	ARM	Raspberry Pi	Crypto Currency Miner	**EnemyBot** [110], [111]	Mar 2022	ARM, MIPS, X86 MISPSel, X64, SH4 M68K, PPC, ARC Sparc	Routers, Modems	Botnet
Reaper/IoTroop [112], [113]	Sep 2017	ARM, MIPS	Routers, CCTV Cameras IP Cameras	Botnet	**Lillin Scanner** [114]	Mar 2022	ARM, MIPS, X86, Sparc M68K, PPC, SH4	Lillin Security Camera DVRs	Botnet
Satori/Okiru [115], [116]	Dec 2017	ARM, MIPS, X86, X64	Huawei Routers	Botnet, DNS Poisoning, Crypto Currency Miner	**RapperBot** [117]	Jun 2022	ARM, MIPS, X86, Sparc	Any Device	Botnet
Hide and Seek [118]	Jan 2018	ARM, MIPS, X86, Sparc, X64, ARC, PPC, SH4, m68k	Any Device	Botnet, Crypto Currency (Monero) Miner, Data Exfiltration	**ZuoRAT** [119]	Jun 2022	MIPS, MIPSel	SOHO Routers	MITM DNS Poisoning
VPNFilter [120], [121]	May 2018	ARM, MIPS X86	Routers, NAS SCADA Control Systems	Information Theft, DNS Poisoning MITM	**Shikitega** [122]	Aug 2022	X86, X64	Any Device	Crypto Cur. Mining (Monero)
Shinoa [123]	May 2018	ARM, MIPS, MISPSel, X86, PPC SH4, Sparc, M68K	Any Device	Botnet	**Chaos** [124]	Sep 2022	ARM, MIPS, X86, PPC	Routers, Modems, Servers	Botnet Crypto Cur. Mining

security sector. Fig. 4 shows the yearly distribution of IoT Malware development to denote the IoT malware growth better.

Some previous researches contain the evolution of IoT malware [9], [125]–[128]. Angrishi presents 14 malware families from Hydra to IRCTelnet on a timeline [125]. On

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

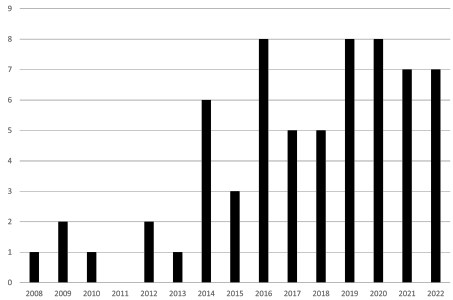


FIGURE 4: Yearly Distribution of IoT Malware Development.

TABLE 5: Inherited Feature Definitions

Feature Number	Feature
1.1	Scanning Module
2.1	Dictionary(Username and Password List)
3.1	Malware Download Server IP Address
3.2	Malware Delivery Technique
4.1	SYN Flood
4.2	ACK Flood
4.3	UDP Flood
4.4	HTTP Flood
4.5	PDoS
5.1	C2 Technique
5.2	Control Server
5.3	C2 Server IP Address
6.1	Obfuscation/Encryption Technique
6.2	Kill Other Botnets or Process

the other hand, Donno et al. also present 13 malware families from Hydra to Mirai on a timeline while showing the relationship between the malware families [126]. Vignau et al. present features of the most commonly seen IoT malware families on the first 10 years of the IoT Malware History and they present a Phylogenic Graph which contains 16 malware families [9]. One of the latest research in this field performed by Ngo et al., the relationship between 12 malware families from Hydra to VPNFilter is presented on their research [127]. Researches [125] and [126] analyzed 13 and 12 malware families from 2008 to 2016 timeframe; in our research, we located 24 malware families for the same timeframe. We believe that the lack of resources causes this difference in the malware family numbers for the newly emerging malware families. Besides, Cozzi et al. presented a lineage of IoT malware families [128]. They benefit from machine learning techniques to find the similarities between the source codes of malware families. Their results were awe-inspiring; most of the malware families in their dataset share common functions. Kim et al. proposed a malware classification based on "Recursive Feature Elimination (RFE)" method [129]. Even though they applied their proposed method only to windows based malware, we believe that their methodology also could be used to classify IoT malware.

When Table IV and Table VII are investigated deeply, it provides insight into the evolution of malware families. For example, from Hydra to Tsunami (2008-2010), mostly MIPS architecture was targeted. After Aidra (2012), other architectures started to become a target for malware families. Also, if the service used for communication with the C2 server under the Management (Mngt) phase of Table IV is investigated, it can be seen that the IRC service is used for C2 communication from Hydra to Tsunami. After Aidra, other services started to be used for communication with the C2 server. This finding indicates that Aidra contains some new features for other architectures, and it has a different module for communicating with C2 Server. In this context, the Phylogenetic Tree of the IoT Malware Families is built-up and shown in Fig. 5. Our Phylogenetic Tree of IoT Malware is an acyclic graph. The nodes contain malware families, and their edges map ancestors. The numbers above the nodes show the inherited features that are defined in Table V.

Our Phylogenetic Tree is created in a time-series manner. It can be observed from our Phylogenetic Tree that the malware development for IoT devices is grown up at a slow pace until 2014, and in 2016 after the infamous Mirai attack, the pace of the IoT malware boomed. From 2008 to 2016 (from Hydra to Mirai), 21 different malware families are located. On the other hand, after Mirai to October 2022, 42 different malware families were located. Also, 29 of these 42 malware families are variants of Mirai, and we assume that there may be dozens of smaller Mirai variants. For this reason, the IoT malware evolution process could be classified as ”Before Mirai” and ”After Mirai.”

One of the critical aspects obtained from the Phylogenetic Tree is the parent malware families. Four different malware families, Hydra, Tsunami, Gafgyt, and Mirai, are determined as parent malware families where some features of these malware families are highly inherited by other malware families. Besides, there are some small branches on the tree. Carna and Wifatch create a small branch as they are classified as "White-Hat Trojans." Also, VPNFilter inherited some features from BrickerBot. On the other hand, there are 12 malware families without any relationship and similarities with other malware families. The inherited features of the malware are discussed below.

A. INHERITED FEATURES FROM HYDRA

Three malware families Psyb0t, Chuck Norris, and Tsunami inherited some features from Hydra. All of these malware families target the same CPU architecture; however, we do not see that as an inherited feature as it is only related to the compile process of malware. The most significant inherited feature is the C2 technique. These malware families used IRC (Internet Relay Chat) channels to send commands to malware [32], [33], [36], [37], [39], [41], [42]. For this reason, these malware families are also known as IRC Botnets. Besides, the "Readme" file of the Hydra malware contains some evidence about the nationality of the developer. The e-mail address of the developer most probably belongs to an Italian person or organization [33]. Moreover, the IP address of the C2 server of Chuck Norris and Tsunami are the same, and the location of this IP address is in Italy [36]. Even though the location

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

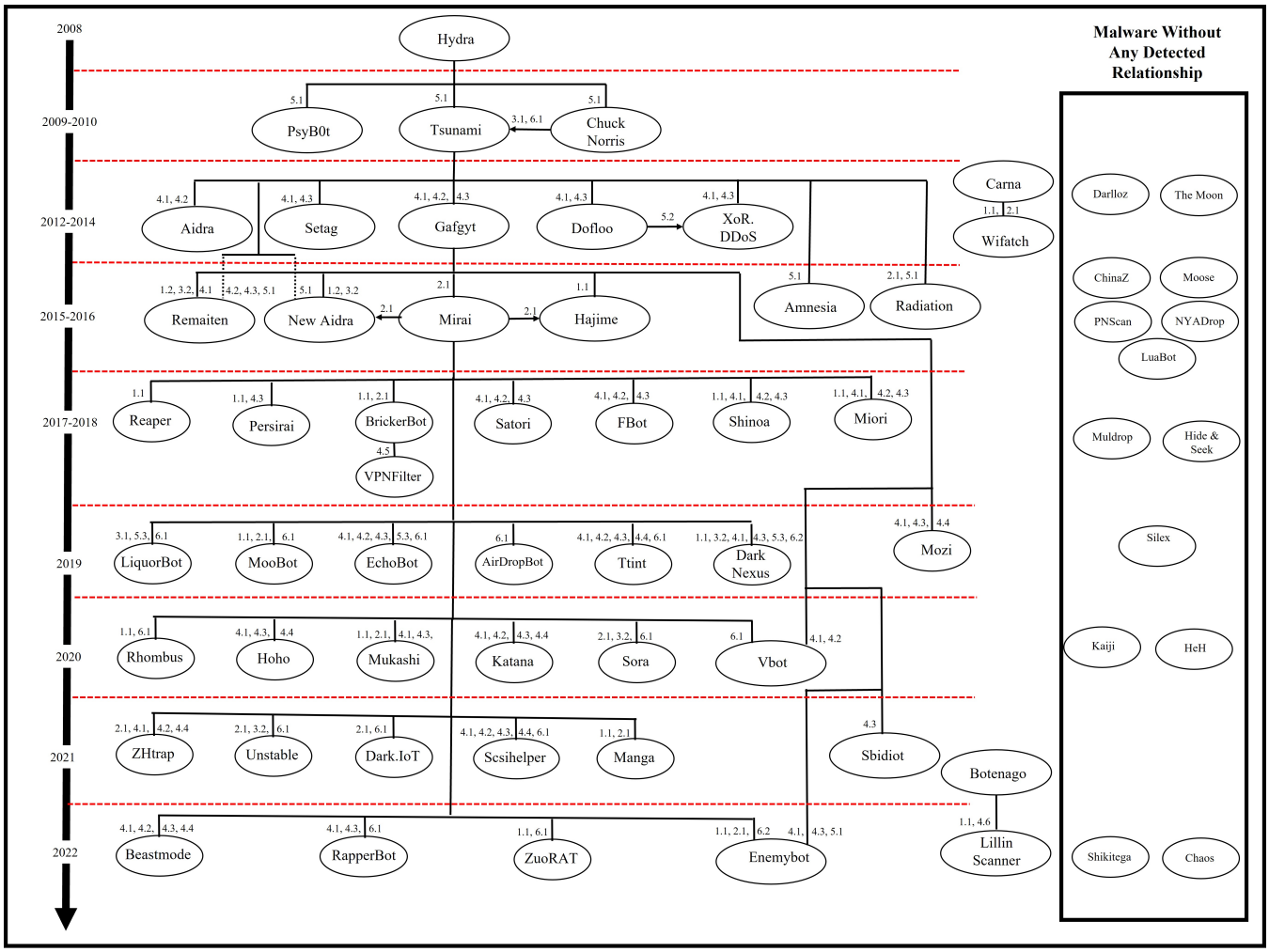


FIGURE 5: Phylogenetic Tree of IoT Malware Families.

of the developers gives us strong evidence of the inspiration from the previous malware, we did not take this feature as an inheritance. However, Tsunami inherited the C2 server and obfuscation and encryption keys from Chuck Norris [36].

B. INHERITED FEATURES FROM TSUNAMI

Seven malware families are located with inherited features from Tsunami. We analyzed the source codes of Tsunami and Aidra to detect the similarities between these two malware [41], [45]. The Aidra has two attack vectors, SYN Flood and ACK Flood. During our source code analysis, we detected that Aidra uses exactly the same source codes with Tsunami for these attack vectors. Reverse engineering efforts of other researchers also showed that Setag, Dofloo, and XoRDDoS malware families inherited SYN and UDP flooding attack vectors of Tsunami. [62], [65], [73]. Besides, XoRDDoS uses a similar control panel with Dofloo which is based on HTTP File Server (HFS) [74]. The successful attack vectors of Tsunami are also adopted by the Gafgyt malware, however more sophisticated initial access and C2 communication techniques differ these malware families from each other [69]. Amnesia and Radiation have different attack vectors than other Tsunami variants, but they have the same C2 technique with Tsunami [86], [91]. Moreover, Radiation has the same dictionary (contains credentials for gaining initial access) with the Tsunami to be used for gaining initial access to the victim machine.

C. INHERITED FEATURES FROM GAFGYT

Eight malware families are located with inherited features from Tsunami. Analysis of the Remaiten revealed the comments of the developer on the source code. According to the comments of the developer, Remaiten adopted the "Telnet Scanning", "Malware Delivery", and "SYN Flooding" modules from Gafgyt; and "ACK Flooding", "UDP Flooding", and "C2 Communication" from Tsunami. [84]. New Aidra also inherited some features from three previous malware. New Aidra adopted the "C2 Communication" module from Tsunami; "Telnet Scanning" and "Malware Delivery", modules from Gafgyt, and it has the same dictionary from Mirai [103]. Hajime also inherits some features from Gafgyt and Mirai. Hajime adopted the scanning module from Gafgyt and

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

the dictionary from Mirai [100], [101]. The infamous Mirai malware was also derived from Gafgyt. The dictionary used in the Gafgyt has 6 usernames and 14 passwords. On the other hand, Mirai has 62 credentials which also include the Gafgyt's set [5]. Mozi, Vbot, and Sbidiot inherit the attack vectors [63], [64], [80], [98], [99]. Enemybot also inherits attack vectors along with C2 module. [110]

D. INHERITED FEATURES FROM MIRAI

As it is mentioned hereinbefore, we are able to locate 29 malware families inspired by Mirai. The key features behind the huge impact of the Mirai attacks can be listed as:

(i) Scanning module for finding new victims,
(ii) Dictionary of credentials to be used for the brute-force attack to gain initial access,
(iii) Simple but effective attack vectors,
(iv) Killing other processes which use Telnet, SSH, and HTTP services to avoid any performance loss and getting detected.
(v) Lightweight obfuscation with encryption and decryption with XoR operation of strings.

Mirai inspired most of the IoT malware families from various aspects. Some of the inspired malware families inherit all of the mentioned features while some of them use only one feature. Reaper only inherits the scanning module [112]. Persirai inherits the scanning module, and UDP flooding [107]. Brickerbot and Manga inherit the scanning module and the dictionary [105], [106]. Satori, FBot, Shinoa, Miori, EchoBot, Ttint, Dark Nexus, Hoho, Katana, ZHtrap, Scsihelper, Beastmode, and RapperBot inherit the DDoS attack vectors of Mirai [34], [38], [51], [52], [57], [59], [60], [75], [85], [89], [90], [102], [108], [116], [117], [123], [130], [131]. Along with DDoS attack vectors, Shinoa and Miori also inherit the scanning module, ZHtrap inherits the dictionary, and Echobot, Ttint, and Scsihelper inherit the obfuscation technique of Mirai. EchoBot also has the same C2 server IP address with Mirai. Dark Nexus also inherits additional features from Mirai, it has the same IP address of C2 and malware download servers and inherits the scanning module and the killing other botnets feature. LiquorBot inherits different features of Mirai. It has the same C2 server, download module, and string obfuscation technique with Mirai [40]. MooBot, Mukashi, Manga, Enemybot inherit the scanning module and dictionary of Mirai [46]–[48], [77], [78], [106], [110], [111]. Additionally, MooBot inherits the obfuscation technique, Mukashi inherits attack vectors, and Enemybot inherits killing other botnets features of the Mirai. Rhombus and Zuorat inherit the scanning module and obfuscation technique while AirdropBot and Vbot only inherit the obfuscation technique with the same XoR key of the Mirai [54], [72], [80], [119]. Sora, Unstable and Dark.IoT inherits the dictionary and obfuscation technique of Mirai [59], [60], [67], [132]. Besides, Sora and Unstable have the same malware download server IP address of Mirai.

VI. BEHAVIOURAL ANALYSIS OF MALWARE FAMILIES

Most of the malware families located in this project's scope follow a common threat pattern. The common threat pattern contains five phases to compromise the target machine. As the first step of the attack, malware scans the network to find a target via its built-in scanning module. In the second phase, the malware tries to establish initial access to the target. There are two different techniques adopted by the malware to succeed in this phase. One of these techniques is trying default password and username combinations in a brute-force attack and the other technique is exploiting known vulnerabilities of a running service on the victim machine. After establishing initial access successfully the third phase of the attack begins. In this phase, the malware communicates with the C2 server or P2P host to download malware ELF files to the victim device. After downloading the malicious ELF file, the malware waits for the fourth phase of the attack. In this phase, malware listens to the dedicated port for commands from the C2 server to conduct malicious activities. The fifth and last phase does not apply to all the malware families in our collection. Some of the malware families try to maintain their presence on the target device or avoid getting detected or analyzed. For this reason, malware families apply different techniques such as killing other malware processes, blocking some ports, or even wiping the target device. The common threat pattern of the IoT malware is given in Fig. 6.

As a result of our efforts, we present the behavioral analysis IoT malware families by adopting a layered investigation framework. By utilizing this framework, the tactics of the adversaries to exploit the target devices, and techniques (along with ID numbers) used for conducting the mentioned tactics are given below and in Table VII. Besides attack vectors, exploited vulnerabilities, communication ports, and services are presented. Our investigation framework of the Behavioral Analysis of IoT Malware Families is shown in Table VII in Appendix A.

A. PLANNING

- **Common Tactic: Discovery:** The attacker tries to collect information about the target device.
- **Techniques:** There are two different techniques to achieve the "Discovery" tactic.
  - **Remote System Discovery (T0846):** Detection of running hosts by IP address, hostname, or other logical identifiers on a network.
  - **Network Service Scanning (T0841):** Conducting port scanning to locate running services on the victim host.

In the planning phase for reconnaissance, all 64 malware families are using the discovery tactic on Mitre ATT&CK for ICS. To conduct this tactic, they used two different techniques which are, "Remote System Discovery" and "Network Service Scanning," regarding their attack vector in the Preparation phase. All the malware families used their built-in scanning modules to accomplish the reconnaissance phase.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

The distribution of the used techniques is shown in Fig. 7.

B. PREPARATION

- **Common Tactic: Initial Access:** The attacker tries to get access to the target device.
- **Common Technique: Internet Accessible Device (T0883):** Attackers gain access to Edge/Fog computing environment through devices exposed directly to the internet.

In the preparation phase, malware is weaponized, and target hosts are set by the result of the previous phase. The used tactic in this phase is the same for all malware families. All the malware families try to establish an "Initial Access" to the victim host. To conduct this tactic, all the malware families used the "Internet Accessible Device" technique to establish a first connection with the victim host. Malware families differ by the attack vectors during this phase. In order to gain initial access to the victim machine, some malware families use publicly known built-in credentials for the Brute-Force attacks. For example, the infamous Mirai malware has a dictionary consisting of 61 credential pairs to brute force the victim device. The other attack vector is exploiting the vulnerabilities of a target device. All the used vulnerabilities are shown in Table VII. While some of the malware families use only one attack vector, others use both attack vectors. The distribution of the used attack vectors is shown in Fig. 8.

C. CYBER INTRUSION

The Cyber Intrusion phase contains two sub-phases as Delivery and Exploitation.

1) Delivery

- **Common Tactic: Persistence:** The attacker tries to get access to the target device.
- **Common Techniques: Program Download (T0843):** Performing a program download to transfer malicious software

All malware families follow a common tactic and technique in this sub-phase. One of the essential capabilities of malware is to be able to achieve persistence. The persistence tactic is used by attackers to maintain their presence in the victim device after gaining initial access in the previous phase. The "Program Download" technique is used to conduct this tactic. Attackers attempt to infect the victim device with a malicious code. There are two different attack vectors in this subphase, after the initial access to the victim device, most of the malware families attackers download their malicious code from their servers to ensure their presence on the system. On the other hand, instead using a centralised malware download server, five of the malware families (Wifatch, Hajime, Hide and Seek, Mozi, and HeH) deliver malware with P2P connection.

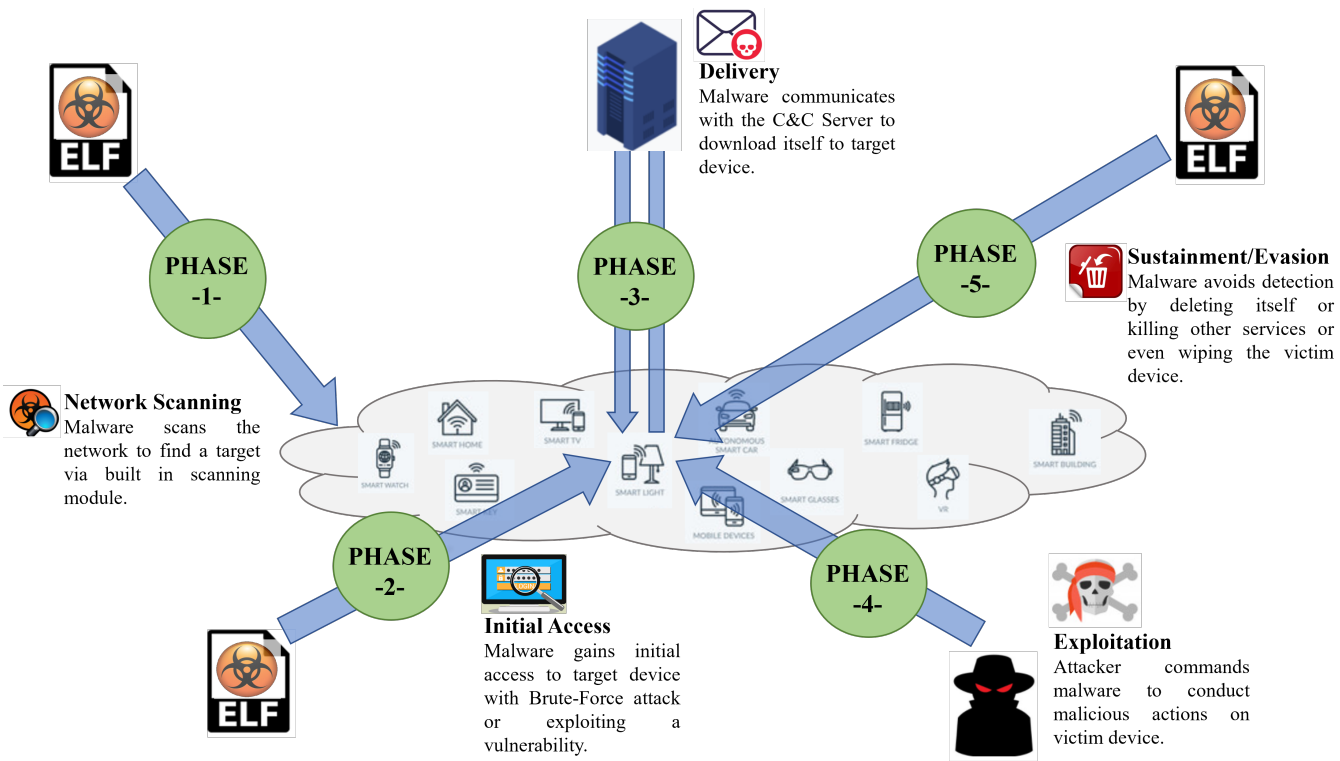


FIGURE 6: Common Threat Pattern of IoT Malware Families.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

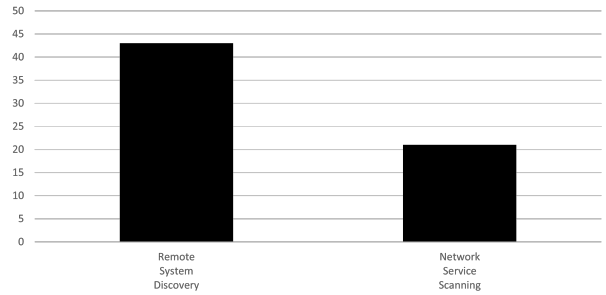


FIGURE 7: Distribution of Techniques on Planning Phase.

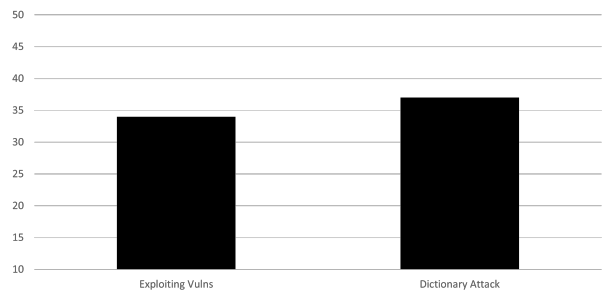


FIGURE 8: Distribution of Attack Vectors on Preparation Phase.

2) Exploitation

- **Tactics:** There are four different tactics in Exploitation Subphase
  - **Inhibit Response Function:** Preventing the target device's safety, protection, quality assurance, and operator intervention functions from responding to a failure, hazard, or unsafe state.
  - **Impact:** Aims to manipulate, interrupt, or destroy target device.
  - **Execution:** Running malicious code on the victim host.
  - **Persistence:** Maintaining presence in the victim device.
- **Techniques:** There are eight different techniques to achieve relevant tactics.

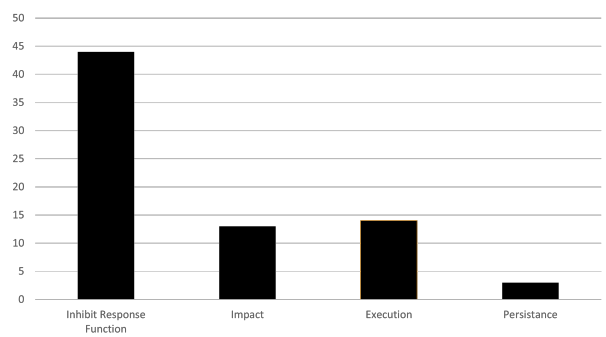


FIGURE 9: Distribution of Tactics on Exploitation Phase.

  - **Denial of Service (T0814):** Performing Denial-of-Service (DoS) attacks to disrupt expected device functionality.
  - **Manipulation of Control (T0831):** Manipulation on communication or commands of victim host.
  - **Loss of View (T0829):** Causing permanent loss of view where the target IoT device requires local, hands-on operator intervention.
  - **Theft of Operational Information (T0882):** Stealing important data.
  - **Loss of Availability (T0826):** Disrupting essential components or systems.
  - **Command Line Interface (T0807):** Utilizing command line interfaces (CLIs) to interact with victim device and execute commands.
  - **Scripting (T0853):** Using scripting languages to execute arbitrary code.
  - **Program Download (T0843):** Performing a program download to transfer malicious software.

In this sub-phase, all malware families differ from each other in terms of Tactics, Techniques, and Attack Vectors. Malware families use four different tactics (Inhibit Response Function, Impact, Execution, and Persistence) to compromise the victim device. The distribution of used Tactics is shown in Fig. 9.

Eight different techniques are used to conduct those tactics. For the Inhibit Response Function tactic, the DDoS technique is used. For the Impact tactic, Manipulation of Control, Loss of View, Theft of Operational Information, and Loss of Availability techniques are used. For the Execution tactic, Command Line Interface and scripting techniques are used. Lastly, for the Persistence tactic, the Program Download technique is used. The distribution of attack vectors by techniques is shown in Fig. 10.

IoT malware families used 18 different attack vectors to exploit the victim device. Eight of these attacks are the "flooding attacks" for DDoS attacks. After the DDoS attacks, the most used attack type is DNS Spoofing. Some malware families like Ttint and LuaBot were detected to be used for SOCKS (Socket Secure). With the newly emerging paradigm Malware-as-a-Service (MaaS), these botnets are rented to malicious people to route their malicious traffic over these victim devices as proxy servers. Another interesting attack vector is cryptocurrency mining. These devices are highly constrained in terms of computation power. For this reason, it is impossible to mine Bitcoin, which requires high computational power. However, there are a few other crypto coins like "Monero," which do not require too much computation power for mining activities. Eight of the malware families (13%) have an attack vector of cryptocurrency mining. There are two malware families, "Chuck Norris" and "Brickerbot," which make the victim devices unusable without operator intervention. Chuck Norris re-configures the device firmware and requires a user to reboot the device. On the other hand, "BrickerBot" wipes the partial or whole of the device

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

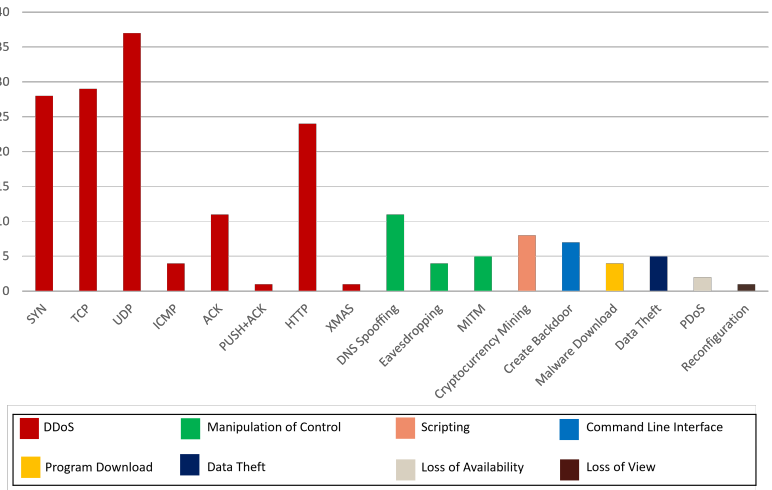


FIGURE 10: The Distribution of Attack Vectors by Techniques on Exploitation Phase.

memory and attack with the "Permanent Denial of Service (PDoS)" attack vector. After the "BrickerBot" attack, more expertise intervention is required to make the device operable again. The other attack vectors are "Download Another Malware," "Social Media Hijacking," "Eavesdropping," "Man in the Middle (MITM)," "Creating Backdoor," and "Executable Installer."

D. MANAGEMENT

- **Common Tactic: Command and Control(C2):** The attacker tries to communicate and control the victim machine to use it for malicious purposes.
- **Techniques:** There are two different techniques to achieve this tactic.
  - **Commonly Used Port (T0885):** Communicating on a commonly used port to not get detected by firewalls or network detection systems and to hide behind normal network activity to avoid more detailed inspection.
  - **Connection Proxy (T0884):** Using a connection proxy to direct network traffic between systems or act as an intermediary for network communications. Also, this definition of a proxy can be expanded to encompass trust relationships for P2P networks consisting of hosts or systems that regularly communicate with each other.

In the management phase, all the malware families adopt the same Tactic. Malware developers use "Command and Control Tactic" to conduct their malicious activities on the infected device. They need to establish communication to send their commands to malware and run this command in the victim device. Two different techniques are applied for achieving this tactic. Most malware families apply the "Commonly Used Ports" technique to hide behind normal network data packets.

On the other hand, five of the malware families apply the "Connection Proxy" technique. Malware families Wifatch, Hajime, Hide and Seek, Mozi, and HeH use P2P network communication to communicate with the infected device. This new technique makes it impossible to centrally disinfect these malware families because malware researchers and security companies generally disable malicious C2 servers. Since P2P botnet malware does not need such a central C2 server, it is much more difficult to disinfect them centrally [133].

All the malware families adopted the same approach in this phase and apply the "Commonly Used Port" technique. After the victim machine gets infected, the malware establishes communication with C2 servers to get commands and download extra malicious files if needed. However, malware families get differ on the attack vector side. Fourteen different ports and seven different services are used by malware to communicate with their server. IRC, Telnet, TCP, UDP, DHT, SSH, and WSS services are used for this purpose. The distribution of the used ports is shown in Fig. 11. On the other P2P IoT botnet malware families do not have any central C2 servers, and their communication ports and services may vary by the command. The details of the communication protocols of the P2P botnets could be found in the relevant analysis researches [63], [87], [118].

E. SUSTAINMENT/EVASION

- **Tactics:** There are five different tactics for sustainment on the target or evasion of the target to avoid detection or removal from the device.
  - **Inhibit Response Function:** Preventing the target device's safety, protection, quality assurance, and operator intervention functions from responding to a

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

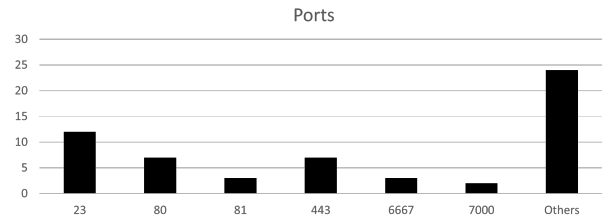


FIGURE 11: Distribution of Used Ports on Management Phase.

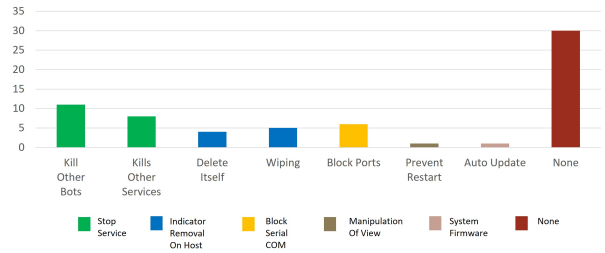


FIGURE 12: Distribution of Attack Vectors by Techniques on Sustainment/Evasion Phase.

failure, hazard, or unsafe state.

- **Impair Process Control:** Trying to manipulate, disable, or damage running processes.
- **Evasion:** Avoiding to get detected.
- **Persistence:** Maintaining presence in the victim device.
- **Impact:** Aims to manipulate, interrupt, or destroy target device.

- **Techniques:** There are five different techniques to achieve relevant tactics.
  - **Block Serial COM (T0805):** Blocking access to serial communication ports to prevent instructions or configurations from reaching target hosts.
  - **Service Stop (T0881):** Stopping or disabling services on a system to render those services unavailable to other users.
  - **Indicator Removal on Host (T0872):** Attempting to remove artifacts of their presence on the target host to cover their tracks.
  - **System Firmware (T0857):** Exploiting the firmware update feature on accessible devices to upload malicious or out-of-date firmware.
  - **Manipulation of View (T0832):** Attempting to manipulate the information reported back to operators or controllers.

As in the Exploitation sub-phase, all malware families differ in Tactics, Techniques, and Attack Vectors in this phase. Five different tactics are used by malware families for sustainment or avoiding detection purposes. It can be observed from Table VII that malware families before 2016 (before Remaiten) do not have any sustainment or evasion tactics generally. Only two of the malware families, Wifatch and Setag, used sustainment tactics. If we take into consideration that Wifatch is a "White-Hat-Trojan" and aims to protect IoT devices from getting infected by other malware, Setag is the first malware that uses sustainment tactics to maintain its presence on the target host. Even though some new malware types emerged between Setag and Remaiten, none of them applied any tactics for sustainment and evasion purposes.

Also, five different techniques are used to conduct those tactics. For the Inhibit Response Function tactic, the "Block Serial COM" technique is used. With this technique, malware blocks some specific ports generally used by other malware. As mentioned hereinbefore, IoT devices are highly constrained devices in terms of computation power, and running more than one malware may bring performance loss. This performance loss may be noticed by the user, and the user may take some prevention against it. With this understanding, this technique is generally used for the prevention of other IoT botnet malware infecting the device. Also, the "Service Stop" technique under the Impair process control is used for the same purpose. With this technique, if there is any other botnet malware on the victim host, it would be killed before beginning any other malicious activity.

Another technique used in this stage is "Indicator Removal on Host" to conduct the "Evasion" tactic. With this technique, the malware tries not to get detected on the victim device. To avoid getting detected, different attack vectors are used by malware families. Remaiten and Liquorbot delete themselves or some of the downloaded malware files from the C2 server. Amnesia has a very particular attack vector. If it detects that it is running in a Virtual Machine, It wipes the VM to avoid being exposed and getting analyzed. Birckerbot, VPNFilter, Silex, and HeH wipe some part or whole of the host device.

Only one malware family uses the System Firmware technique to conduct the Persistence tactic. VBot has a feature to update itself automatically, like a firmware update. Another technique used by only one malware is the Manipulation of View. This technique is applied only by Katana to conduct the “Impact” tactic. This malware prevents the infected device from restarting to maintain its existence. The distribution of attack vectors by techniques is shown in Fig. 12.

F. LESSONS LEARNED

Defense-in-depth approach needs to be applied to defend against IoT malware. An efficient defense mechanism should aim to defend the whole computing environment for every phase of the malware attack. In the first phase of the attack, two different attack vectors are applied by the malware. Both of these attack vectors are based on network scanning. One of these attack vectors is discovering the running hosts on the network and the other one is discovering running services on these hosts. Defending against both of these attack vectors is very easy by configuring a firewall properly.

The second phase also contains two different attack vec-

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

TABLE 6: The Defense-In-Depth Approach For IoT Malware

Phase		Attack Vector	Defence Mechanism
Planning		System Discovery	Firewall Configuration
		Service Scanning	
Preparation		Dictionary Attack	Changing Default Credentials
		Exploit Vulnerabilities	Patching System
Cyber Intrusion	Delivery	Download Malware From Server	Blocking The IP Address Of Malware Download Servers
			Anti-Malware Tools (Static Analysis)
	Exploitation	Attack Vectors	Firewall Configuration
			Anti-Malware Tools (Dynamic Analysis)
Management		C2 Server Communication	Blocking The IP Address Of C2 Servers
			Blocking Unnecessary Ports
Sustainment		Attack Vectors	Anti-Malware Tools (Dynamic Analysis)

tors. Defense against these attack vectors needs the awareness of the users. Most of the malware uses default username and password tuples to gain initial access. Changing default usernames and passwords can easily provide security against these malware families. The other attack vector is exploiting vulnerabilities to gain initial access. To defend against these malware families, the best action is to apply patches in a timely manner.

The third phase contains two sub-phases. The first sub-phase is delivery. Blocking the known IP address of known malware download servers may provide security partly. The other solution is deploying anti-malware tools based on static analysis. Nevertheless, there are no prevailing anti-malware tools for IoT devices. The second sub-phase is the exploitation sub-phase. There are two different defense solutions for this phase. One of these solutions is a proper firewall configuration. The other solution is again deploying an anti-malware tool, but in this sub-phase, the tool should be based on the dynamic analysis of malware and to the best of our knowledge there is no such anti-malware tool that runs on IoT devices for now.

In the fourth phase, the applicable defense mechanism is very similar to the delivery sub-phase. Blocking the IP address of the known C2 servers prevents communication between the malware and the C2 server. Also blocking unnecessary ports is also an efficient solution.

The last phase contains the sustainment techniques of the malware. Like the exploitation sub-phase, a dynamic analysis based anti-malware tool should be used to detect sustainment techniques, such as killing other processes, blocking some ports, or deleting some files. The defense-in-depth approach for IoT malware is given in Table VI.

VII. CONCLUSION

IoT malware poses a severe threat to all devices in Edge/Fog computing environment. These ecosystems are still far from being secure, and there is much progress that needs to be done for securing edge devices. Serious steps have to be taken for developing efficient defense mechanisms against malware for edge devices. We believe that developing an effective anti-malware mechanism is possible only with a high understanding of malware behaviors. This paper aims to bring light for future researches with presented features of the IoT malware.

This research fills three important research gaps for an improved understanding of defending IoT devices against malware attacks. As a first step, 64 different malware families have been introduced to familiarise IoT malware. This 64 different malware families cover the majority, if it is not all, of existing IoT malware ranging from its debut in February 2008 to recent ones by the published date of this paper. Then, the evolutionary development of these malware families are presented with a phylogenetic tree. Four different malware families Hydra, Tsunami, Gafgyt, and Mirai, are determined as parent malware families. Those malware families could be assessed as milestones in IoT Malware development and their features are highly inherited by other malware families. Lastly, a methodical characterization of existing IoT malware families is presented. The malware families are characterized by various aspects, such as target architecture, evolution, delivery techniques, attack vectors, and sustainment methods. Further detailed analysis of IoT malware shows that the velocity of newly emerging malware families, increased sophistication, and newly applied techniques such as P2P communication and obfuscation techniques pose significant challenges for IoT malware detection.

Considering the significant malware threat for IoT devices, it is surprising to see that the literature lacks a fundamental methodology to guide malware analysis and development of anti-malware tools in the IoT domain. We believe that the first step for securing IoT devices highly depends on the understandability of the malware behaviors. During the literature review, we surprisingly realize that there is only a limited number of researches on the IoT malware analysis domain. Most of the researches focused on malware detection based on machine learning. Developing efficient anti-malware tools needs to combine the efforts of two different disciplines which are Malware Analysis and Machine Learning. These two disciplines require different skills. Specifically, malware analysis mostly requires reverse engineering skills along with knowledge of operating systems and network communication, while machine learning requires mostly statistics. In this regard, a malware analysis framework that combines static and dynamic analysis methods is highly needed for providing a dataset for the training process of machine learning-based approaches.

Despite the novel approach of the study, it is worth mentioning some remaining challenges and limitations. Securing edge devices against malware is insufficient for providing

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

a secure ecosystem in Edge/ Fog computing. This research only presents the malware threat on the edge devices layer. Similar research also has to be conducted for the other layers of the Edge/Fog computing environment. Providing end-to-end security on all over the Edge/Fog Computing ecosystem is only possible by securing all the layers.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

APPENDIX A MALWARE BEHAVIORAL ANALYSIS FRAMEWORK FOR EDGE DEVICES

TABLE 7:
Behavioral Analysis of IoT Malware Families

Malware Family	Planning & Preparation			Cyber Intrusion				Management		Sustainment/Evasion		
	Reconnaissance		Weaponization/Targetting	Delivery	Exploitation			C2				
	Common Tactic:	Discovery	Common Tactic: Initial Access Common Technique: Internet Accessible Device	Common Tactic: Persistance Common Technique: Program Download				Common Tactic:	Command and Control			
	Techniques	Attack Vector	Attack Vector	Attack Vector	Tactics	Techniques	Attack Vector	Techniques	Port & Service	Tactics	Techniques	Attack Vector
Hydra	Remote System Discovery	Auto Scan	Default Credentials, D-Link Password Extraction Vulnerability (CVE -2018-7034)	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP Flood	Commonly Used Port	6667 IRC	-	-	-
Psyb0t	Remote System Discovery	Auto Scan	Authentication Bypass (cgi-bin/ firmwarecfg)	Download Mal. Software From Server	Inhibit Response Function, Impact	DDoS, Manipulation of Control	SYN, UDP, ICMP Flood Download Arbitrary File, Visiting Webpage	Commonly Used Port	23 IRC	-	-	-
Chuck Norris	Remote System Discovery	Auto Scan	Telnet Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function, Impact	DDoS, Manipulation of Control, Loss of View	SYN Flood, DNS Spoofing, Device Reconfiguration	Commonly Used Port	6667 IRC	-	-	-
Tsunami/ Kaiten	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function, Impact	DDoS, Manipulation of Control	SYN, UDP, Push+ACK Flood, DNS Spoofing	Commonly Used Port	6667 IRC	-	-	-
Aidra/ L. Aidra Zendran	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, ACK Flood	Commonly Used Port	80 tcp	-	-	-
Carna	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Impair Process Control	Brute-Force I/O	White-Hat Trojan	Commonly Used Port	23 Telnet	-	-	-
Darlloz	Remote System Discovery	Auto Scan	Dictionary Attack, "php-cgi" Information Disclosure Vulnerability (CVE-2012-1823)	Download Mal. Software From Server	Execution	Scripting	Crypto Currency Mining	Commonly Used Port	23 Telnet	-	-	-
Wifatch	Remote System Discovery	Auto Scan	Dictionary Attack	P2P Mal. Software transfer	Impair Process Control	Brute-Force I/O	White-Hat Trojan	Connection Proxy	23 Telnet	Inhibit Response Function	Block Serial Com	Block Telnet Ports
TheMoon	Remote System Discovery	Auto Scan	XML HNAP Vulnerability	Download Mal. Software From Server	Unknown	Unknown	Unknown	Commonly Used Port	80, 8080 TCP	-	-	-
Setag/ Mayday/ Elknot/ BillGates	Remote System Discovery	Auto Scan	Groovy Scripting Engine Bypass Protection Vuln. (CVE-2015-1427)	Download Mal. Software From Server	Inhibit Response Function, Execution	DDos, Scripting	SYN, TCP, UDP, ICMP, HTTP Flood, Crypto Currency Mining	Commonly Used Port	81 TCP	Impair Process Control	Service Stop	Kills Other Mining Services
Dofloo/ Spike/ Mr Black DnsAmp	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, LSYN, UDP, UDPS, TCP Flood	Commonly Used Port	2375 TCP	-	-	-
Gafgyt/ Bashlite/ Lizkebab/ Torlus	Remote System Discovery	Auto Scan	Remote Code Execution Vulnerabilities (CVE-2014-8361), (CVE-2017-17217), (CVE-2017-18368), (CVE-2018-15887)	Download Mal. Software From Server	Inhibit Response Function, Impact	DDoS, Manipulation of Control	SYN, UDP, ACK, HTTP Flood, DNS Spoofing	Commonly Used Port	23 Telnet	-	-	-
XoR.DDoS	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, TCP Flood	Commonly Used Port	443 SSH	-	-	-
ChinaZ/ Zaniach	Remote System Discovery	Auto Scan	ShellShock Vulnerability (CVE-2014-6271)	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, ICMP, DNS Flood	Commonly Used Port	9521 TCP	-	-	-
Moose	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Impact, Inhibit Response Function,	Data Theft, Manipulation of Control	Social Media Hijacking, DNS Spoofing, MITM	Commonly Used Port	23 Telnet	-	-	-
PNScan	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, ACK Flood	Commonly Used Port	9000 Unknown	-	-	-

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

TABLE 7:
Behavioral Analysis of IoT Malware Families

Malware Family	Planning & Preparation			Cyber Intrusion				Management		Sustainment/Evasion		
	Reconnaissance		Weaponization/Targetting	Delivery	Exploitation			C2				
	Common Tactic:	Discovery	Common Tactic: Initial Access Common Technique: Internet Accessible Device	Common Tactic: Persistance Common Technique: Program Download				Common Tactic:	Command and Control			
	Techniques	Attack Vector	Attack Vector	Attack Vector	Tactics	Techniques	Attack Vector	Techniques	Port & Service	Tactics	Techniques	Attack Vector
Remaiten	Network Service Scanning	Auto Telnet Scan	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function, Impact	DDoS, Manipulation of Control	SYN, UDP, ACK, TCP, HTTP Flood, DNS Spoofing	Commonly Used Port	47823 IRC	Impair Process Control, Evasion	Service Stop, Indicator Removal on Host	Kills Other Bots, Deletes Itself.
Amnesia	Network Service Scanning	Auto CCTV Scan	TVT Digital DVRs RCE Vulns. (EDB ID:39596)	Download Mal. Software From Server	Inhibit Response Function	DDoS	UDP, HTTP Flood	Commonly Used Port	443 IRC	Impair Process Control, Evasion	Service Stop, Indicator Removal on Host	Kills Other Bots, VM Wiping
NYA.Drop	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Execution, Persistance	Command Line Interface, Program Download	Create Backdoor, Install Downloader For Other Malware	Commonly Used Port	Unknown	-	-	-
Radiation	Remote System Discovery	Auto Scan	Dictionary Attack CCTV RCE Vulnerabilities (EDB ID:39596), [CVE-2014-6271)	Download Mal. Software From Server	Inhibit Response Function	DDoS	UDP, TCP, HTTP Flood	Commonly Used Port	443 IRC	Impair Process Control	Service Stop	Kills Other Bots
LuaBot	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function, Impact	DDoS, Manipulation of Control	HTTP Flood, DNS Spoofing, MITM, Act as SOCKS	Commonly Used Port	80 TCP 1085 HTTP	-	-	-
Mirai	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, ACK, TCP HTTP Flood	Commonly Used Port	23 Telnet	Impair Process control	Service Stop	Kills All Telnet, SSH, HTTP Using Process
Hajime	Remote System Discovery	Auto Scan	Telnet Dictionary Attack, Tr-069 Vuln., Arris Cable Modem Password of the Day Vulnerability	P2P Mal. Software Transfer	Unknown	Unknown	Unknown	Connection Proxy	1457 UDP, TCP	Inhibit Response Function	Block Serial Com	Block Ports: 53, 5358, 55555, 7547
IRCTelnet New Aidra	Remote System Discovery	Auto Scan	Telnet Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, ACK, TCP Xmas Flood	Commonly Used Port	23 Telnet	Impair Process Control	Service Stop,	Kills Other Bots
BrickerBot	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Impact, Inhibit Response Function	Loss of Availability, DDoS	PDoS	Unknown	Unknown	Evasion	Indicator Removal on Host	Wipe Host Device
Persirai	Network Service Scanning	Auto Scan (Port 81)	Universal Plug And Play (UPnP) Vulnerability	Download Mal. Software From Server	Inhibit Response Function	DDoS	UDP Flood	Commonly Used Port	81 UDP	-	-	-
MulDrop	Network Service Scanning	Auto Scan (Port 22)	Default Credentials (username=pi pass=raspberry	Download Mal. Software From Server	Execution	Scripting	Crypto Cur Mining	Unknown	Unknown	-	-	-
Reaper/ IoTroop	Remote System Discovery	Auto Scan	Authentication Vulns of D-link, GoAhead, JAWS, Netgear, Vacron, LiinkSYS, AVTECH (CVE-2017-8225)	Download Mal. Software From Server	Execution, Persistance	Command Line Interface, Program Download	Create Backdoor, Install Downloader For Other Malware	Commonly Used Port	81 TCP	-	-	-
Satori/ Okuri/	Remote System Discovery	Auto Scan	Realtek SDK Miniigd UPnP SOAP Command Execution Vuln. (CVE-2014-8361) Huawei Home Gateway RCE Vuln (CVE-2017-17215)	Download Mal. Software From Server	Execution Persistance	Scripting, Command Line Interface, Program Download	Crypto Currency Mining, Create Backdoor, Downloader For Other Malware	Commonly Used Port	7000 TCP	-	-	-
Hide&Seek	Remote System Discovery	Auto Scan	Dictionary Attack, Authentication Vulns of Netgear, TP-Link, ZyXEL Routers (CVE-2016-10401)	P2P Mal. Software Transfer	Inhibit Response Function, Impact, Execution	DDoS, Data Theft, Scripting	TCP, UDP, HTTP Flood, Data Theft, Crypto Currency. (Monero) Mining	Connection Proxy	Port Vary IAW P2P Command UDP	Impair Process Control	Service Stop	Kills Other Bots
VPNFilter	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Impact, Inhibit Response Function	Data Theft, Manipulation of Control Loss of Availability	Data Theft, DNS Spoof MITM, PDoS	Commonly Used Port	80, 443 TCP	Evasion	Indicator Removal on Host	Writes Data Over OS
Fbot	Network Service Scanning	Auto Scan (Port 5555)	ADB Service Vuln	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, ACK, TCP HTTP Flood	Commonly Used Port	7000 TCP	Impair Process Control	Service Stop,	Kills Other Bots

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

TABLE 7:
Behavioral Analysis of IoT Malware Families

Malware Family	Planning & Preparation			Cyber Intrusion				Management		Sustainment/Evasion		
	Reconnaissance		Weaponization/Targetting	Delivery	Exploitation			C2				
	Common Tactic:	Discovery	Common Tactic: Initial Access Common Technique: Internet Accessible Device	Common Tactic: Persistance Common Technique: Program Download				Common Tactic:	Command and Control			
	Techniques	Attack Vector	Attack Vector	Attack Vector	Tactics	Techniques	Attack Vector	Techniques	Port & Service	Tactics	Techniques	Attack Vector
LiquorBot	Network Service Scanning	Auto SSH Scan	Dictionary Attack, Exploit Critical Vulnerabilities (CVE-2013-3568), (CVE-2015-2051) (CVE-2016-1555), (CVE-2016-6277) (CVE-2016-6579), (CVE-2017-6077) (CVE-2017-6334), (CVE-2017-6884) (CVE-2018-9285), (CVE-2018-10532) (CVE-2018-17173), (CVE-2019-12780)	Download Mal. Software From Server	Execution	Scripting	Crypto Cur (Monero) Mining	Commonly Used Port	7630, 3333 TCP	Evasion	Indicator Removal on Host	Deletes Dropped Files
Silex	Network Service Scanning	Auto Telnet Scan	Dictionary Attack	Download Mal. Software From Server	Impact, Inhibit Response Function	Loss of Availability, DDoS	PDoS	Unknown	Unknown	Evasion	Indicator Removal on Host	Wipe Host Device
AirdropBot	Network Service Scanning	Auto Scan (Port 8080)	LinkSYS cgi Vuln. EDB ID:43013	Download Mal. Software From Server	Inhibit Response Function	DDoS	TCP, UDP, Flood	Commonly Used Port	455 UDP	Impair Process Control	Service Stop	Kills Other Bots
EchoBot	Remote System Discovery	Auto Scan	Mitsubishi Electric's Remote Terminal Unit (RTU) Vuln. (CVE-2019-14927)	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, ACK, TCP HTTP Flood	Commonly Used Port	23 Telnet	-	-	-
Ttint	Remote System Discovery	Auto Scan	Tenda Router 0-Day Vulns. (CVE-2018-14558) (CVE-2020-10987)	Download Mal. Software From Server	Inhibit Response Function, Impact	DDoS, Manipulation of Control	SYN, UDP, ACK, TCP HTTP Flood, Data Theft, DNS Spoof, Eavesdrop, MITM, Act as SOCKS	Commonly Used Port	443 WSS	Inhibit Response Function	Block Serial Com	Block SSH and Telnet ports.
DarkNexus	Network Service Scanning	Auto Telnet Scan	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function	DDoS	UDP, TCP, HTTP Flood	Commonly Used Port	12000 13000 30047 UDP, TCP	Impair Process Control	Service Stop	Kills Other Bots
Mozi	Network Service Scanning	Auto Telnet Scan	Dictionary Attack, Authentication Vulnerabilities (CVE-2014-8361), (CVE-2017-17215), (CVE-2018-10561), (CVE-2018-10562)	P2P Mal. Software Transfer	Inhibit Response Function	DDoS	SYN, UDP, TCP, HTTP Flood	Connection Proxy	6881 DHT	Inhibit Response Function	Block Serial Com	Block SSH and Telnet ports.
Sora	Remote System Discovery	Auto Scan	Dictionary Attack, (CVE-2017-17215), (CVE-2018-10561), (CVE-2020-6756)	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, TCP, HTTP Flood	Commonly Used Port	23 Telnet	-	-	-
Rhombus	Network Service Scanning	Auto Telnet Scan	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function	DDoS	UDP, TCP, HTTP Flood	Commonly Used Port	2020 TCP	Impair Process Control Control	Service Stop	Kills Other Bots
Mukashi	Network Service Scanning	Auto Scan (Port 23)	Dictionary Attack, ZyXEL NAS Pre-authentication Command Injection Vulnerability (CVE-2020-9054)	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, ACK, TCP HTTP Flood	Commonly Used Port	348, 4864 UDP	Impair Process Control	Service Stop	Kills All Telnet, SSH, HTTP Using Process
Vbot	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function	DDoS	UDP, TCP, Flood	Commonly Used Port	2017 TCP	Persistance	System Firmware	Auto Update
Kaiji	Remote System Discovery	Auto Scan	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function	DDoS	UDP, TCP, SYN, ACK Flood	Unknown	Unknown	-	-	-
Katana	Remote System Discovery	Auto Scan	RCE and Code Injection Vulns	Download Mal. Software From Server	Inhibit Response Function	DDoS	UDP, TCP, HTTP Flood	Unknown	Unknown	Inhibit Response Function, Impact	Block Serial Com, Manipulation of View	Block Ports: 53168, 57913 59690, 62471 63749, Prevent Device Restart
HeH	Network Service Scanning	Auto Telnet Scan	Dictionary Attack	P2P Mal. Software Transfer	Execution	Command Line Interface	Create Backdoor	Connection Proxy	Port Vary IAW P2P Command UDP	Impair Process Control, Evasion	Service Stop, Indicator Removal on Host	Kills Process Running on Ports: 22, 23, 69, 80, 81, 420, 1137, 4269, 6969, 7331, 8080, 8443, 9009, 9696, 50023; Wipe Host Device

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

TABLE 7:
Behavioral Analysis of IoT Malware Families

Malware Family	Planning & Preparation			Cyber Intrusion				Management		Sustainment/Evasion		
	Reconnaissance		Weaponization/Targetting	Delivery	Exploitation			C2				
	Common Tactic:	Discovery	Common Tactic: Initial Access Common Technique: Internet Accessible Device	Common Tactic: Persistance Common Technique: Program Download				Common Tactic:	Command and Control			
	Techniques	Attack Vector	Attack Vector	Attack Vector	Tactics	Techniques	Attack Vector	Techniques	Port & Service	Tactics	Techniques	Attack Vector
ZHtrap	Network Service Scanning	Auto Scan (Port 23)	Dictionary Attack, RCE Vulnerabilities (MVPower DVR, NETGEAR, CCTV DVR RCEs, (CVE-2014-8361)	Download Mal. Software From Server	Inhibit Response Function	DDoS	UDP, TCP, HTTP Flood	Commonly Used Port	3000 TOR	Impair Process Control	Service Stop	Prevent Creation Of New Process
Unstable	Remote System Discovery	Auto Scan	Dictionary Attack, ThinkPHP Vulnerability, (CVE-2017-17215), (CVE-2018-10561), (CVE-2020-6756)	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, TCP, HTTP Flood	Commonly Used Port	23 Telnet	–	–	–
Dark.IoT	Network Service Scanning	Auto Scan	(CVE-2021-20090),(CVE-2021-35395) BlackArmor NAS vulnerability	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, TCP, HTTP Flood	Commonly Used Port	443 SSH	Impair Process Control	Service Stop	Kills All Bot Like Process
Sbidiot	Remote System Discovery	Auto Scan	ZTE Routers RCE vulnerability	Download Mal. Software From Server	Inhibit Response Function	DDoS	TCP, UDP, HTTP Flood	Commonly Used Port	80 TCP	Impair Process Control	Service Stop	Kills All Bot Like Process
Scsihelper	Remote Service Scanning	Auto Scan	Gitlab Vulnerability (CVE-2021-22204))	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, ACK, TCP HTTP Flood	Commonly Used Port	80 TCP	Evasion	Indicator Removal on Host	Deletes Itself Detects VM
BotenaGo	Remote Service Scanning	Auto Scan	(CVE-2020-8658), (CVE-2020-8958) (CVE-2020-10173)	Download Mal. Software From Server	Execution	Command Line Interface	Create Backdoor	Commonly Used Port	19412, 31421	–	–	–
Manga	Remote Service Scanning	Auto Scan	(CVE-2021-41653)	Download Mal. Software From Server	Execution	Command Line Interface	Create Backdoor	Commonly Used Port	1024 TCP	Inhibit Response Function	Block Serial Com	Blocks Commonly Used Ports by Botnets
BeastMode	Remote Service Scanning	Auto Scan	(CVE-2022-26210), (CVE-2022-26186), (CVE-2022-25075), (CVE-2021-45382), (CVE-2021-4505), (CVE-2017-17215)	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, ACK, TCP HTTP Flood	Commonly Used Port	80 TCP	–	–	–
Enemybot	Remote System Discovery	Auto Scan (SSH/Telnet)	Dictionary Attack, (CVE-2022-25075 to 25084), (CVE-2020-17456),(CVE-2021-44228/45046), (CVE-2018-10823), (CVE-2022-27226)	Download Mal. Software From Server	Inhibit Response Function	DDoS	SYN, UDP, TCP, HTTP ICMP Flood DNS Amplification SSL/TLS Attack	Commonly Used Port	23 Telnet	Impair Process Control	Service Stop	Kills Other Bots
Lillin Scanner	Remote Service Scanning	Auto Scan (Lillin Devices))	Dictionary Attack	Download Mal. Software From Server	Execution	Command Line Interface	Create Backdoor	Commonly Used Port	Specified During Attack	–	–	–
RapperBot	Remote System Discovery	Auto Scan (SSH Service)	Dictionary Attack	Download Mal. Software From Server	Inhibit Response Function	DDoS	UDP, TCP Flood	Commonly Used Port	443, 5123 TCP	–	–	–
ZuoRAT	Remote Service Sccanning	Auto Scan	(CVE-2020-26878),(CVE-2020-26879)	Download Mal. Software From Server	Impact, Inhibit Response Function	Data Theft, Manipulation of Control	Data Theft, DNS Spoof MITM	Commonly Used Port	48101, 55556, 39500	Impair Process Control	Service Stop	Kills All Process Running on C2 Port
Shikitega	Unknown	Unknown	(CVE-2021-3493),(CVE-2021-4034)	Download Mal. Software With Three Stages	Execution	Scripting	Crypto Currency Mining	Commonly Used Port	Specified During Attack	–	–	–
Chaos	Remote Service Scanning	Auto Scan	(CVE-2017-17215),(CVE-2022-30525)	Download Mal. Software From Server	Execution, Inhibit Response Function	Scripting, DDoS	Crypto Currency Mining, UDP, TCP, SYN Floods	Commonly Used Port	Specified During Attack	–	–	–

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

REFERENCES

[1] Cisco, "Cisco annual internet report (2018–2023) white paper," Cisco, Tech. Rep., March 2020. [Online]. Available: https://www.cisco.com/c/en/us/solutions/executive-perspectives/annual-internet-report/index.html

[2] J. Pan and J. McElhannon, "Future edge cloud and edge computing for internet of things applications," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 439–449, 2017.

[3] SonicWall Inc., "Mid-year update: 2022 sonicwall cyber threat report," SonicWall Inc., Milpitas, CA USA, Tech. Rep., July 2022, [Accessed October 11, 2022]. [Online]. Available: https://www.sonicwall.com/2022-cyber-threat-report/

[4] N. Woolf, "Ddos attack that disrupted internet was largest of its kind in history, experts say," October 2016. [Online]. Available: https://www.theguardian.com/technology/2016/oct/26/ddos-attack-dyn-mirai-botnet, [Accessed February 15, 2021].

[5] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis *et al.*, "Understanding the mirai botnet," in *26th {USENIX} security symposium ({USENIX} Security 17)*, 2017, pp. 1093–1110.

[6] Z. Gittins and M. Soltys, "Malware persistence mechanisms," *Procedia Computer Science*, vol. 176, pp. 88–97, 2020, knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 24th International Conference KES2020.

[7] R. Al-Shaer, J. M. Spring, and E. Christou, "Learning the associations of mitre att amp; ck adversarial techniques," in *2020 IEEE Conference on Communications and Network Security (CNS)*, 2020, pp. 1–9.

[8] A. Wang, R. Liang, X. Liu, Y. Zhang, K. Chen, and J. Li, "An inside look at iot malware," in *International Conference on Industrial IoT Technologies and Applications*. Springer, 2017, pp. 176–186.

[9] B. Vignau, R. Khoury, and S. Hallé, "10 years of iot malware: A feature-based taxonomy," in *2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, 2019, pp. 458–465.

[10] B. Vignau, R. Khoury, S. Hallé, and A. Hamou-Lhadj, "The evolution of iot malwares, from 2008 to 2019: Survey, taxonomy, process simulator and perspectives," *Journal of Systems Architecture*, p. 102143, 2021.

[11] O. Alrawi, C. Lever, K. Valakuzhy, R. Court, K. Snow, F. Monrose, and M. Antonakakis, "The circle of life: A Large-Scale study of the IoT malware lifecycle," in *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, Aug. 2021, pp. 3505–3522. [Online]. Available: https://www.usenix.org/conference/usenixsecurity21/presentation/alrawi-circle

[12] Y. Pa, S. Suzuki, K. Yoshioka, T. Matsumoto, T. Kasama, and C. Rossow, "{IoTPOT}: Analysing the rise of {IoT} compromises," in *9th USENIX Workshop on Offensive Technologies (WOOT 15)*, 2015.

[13] A. Azmoodeh, A. Dehghantanha, and K.-K. R. Choo, "Robust malware detection for internet of (battlefield) things devices using deep eigenspace learning," *IEEE transactions on sustainable computing*, vol. 4, no. 1, pp. 88–95, 2018.

[14] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, and Y. Elovici, "N-baiot—network-based detection of iot botnet attacks using deep autoencoders," *IEEE Pervasive Computing*, vol. 17, no. 3, pp. 12–22, 2018.

[15] S. Garcia, A. Parmisano, and J. Erquiaga, Maria, "Iot-23: A labeled dataset with malicious and benign iot network traffic (version 1.0.0)," 2020. [Online]. Available: https://www.stratosphereips.org/datasets-iot23

[16] M. Safaei Pour, A. Mangino, K. Friday, M. Rathbun, E. Bou-Harb, F. Iqbal, S. Samtani, J. Crichigno, and N. Ghani, "On data-driven curation, learning, and analysis for inferring evolving internet-of-things (iot) botnets in the wild," *Computers & Security*, vol. 91, p. 101707, 2020.

[17] T. Trajanovski and N. Zhang, "An automated and comprehensive framework for iot botnet detection and analysis (iot-bda)," *IEEE Access*, vol. 9, pp. 124 360–124 383, 2021.

[18] M. Botacin, F. Ceschin, R. Sun, D. Oliveira, and A. Grégio, "Challenges and pitfalls in malware research," *Computers & Security*, vol. 106, p. 102287, 2021.

[19] H. HaddadPajouh, A. Dehghantanha, R. Khayami, and K.-K. R. Choo, "A deep recurrent neural network based approach for internet of things malware threat hunting," *Future Generation Computer Systems*, vol. 85, pp. 88–96, 2018.

[20] J. Su, D. V. Vasconcellos, S. Prasad, D. Sgandurra, Y. Feng, and K. Sakurai, "Lightweight classification of iot malware based on image recognition," in *2018 IEEE 42Nd annual computer software and applications conference (COMPSAC)*, vol. 2. IEEE, 2018, pp. 664–669.

[21] H. Alasmary, A. Khormali, A. Anwar, J. Park, J. Choi, A. Abusnaina, A. Awad, D. Nyang, and A. Mohaisen, "Analyzing and detecting emerging internet of things malware: A graph-based approach," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8977–8988, 2019.

[22] E. M. Dovom, A. Azmoodeh, A. Dehghantanha, D. E. Newton, R. M. Parizi, and H. Karimipour, "Fuzzy pattern tree for edge malware detection and categorization in iot," *Journal of Systems Architecture*, vol. 97, pp. 1–7, 2019.

[23] H. Darabian, A. Dehghantanha, S. Hashemi, S. Homayoun, and K.-K. R. Choo, "An opcode-based technique for polymorphic internet of things malware detection," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 6, p. e5173, 2020.

[24] D. Vasan, M. Alazab, S. Venkatraman, J. Akram, and Z. Qin, "Mthael: Cross-architecture iot malware detection based on neural network advanced ensemble learning," *IEEE Transactions on Computers*, vol. 69, no. 11, pp. 1654–1667, 2020.

[25] H. Haddadpajouh, A. Mohtadi, A. Dehghantanaha, H. Karimipour, X. Lin, and K.-K. R. Choo, "A multikernel and metaheuristic feature selection approach for iot malware threat hunting in the edge layer," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4540–4547, 2021.

[26] J. Jeon, J. H. Park, and Y.-S. Jeong, "Dynamic analysis for iot malware detection with convolution neural network model," *IEEE Access*, vol. 8, pp. 96 899–96 911, 2020.

[27] V. Rey, P. M. S. Sánchez, A. H. Celdrán, and G. Bovet, "Federated learning for malware detection in iot devices," *Computer Networks*, vol. 204, p. 108693, 2022.

[28] L. A. Goldberg, P. W. Goldberg, C. A. Phillips, and G. B. Sorkin, "Constructing computer virus phylogenies," *Journal of Algorithms*, vol. 26, no. 1, pp. 188–208, 1998.

[29] K. Allix, Q. Jérome, T. F. Bissyandé, J. Klein, R. State, and Y. Le Traon, "A forensic analysis of android malware–how is malware written and how it could be detected?" in *2014 IEEE 38th Annual Computer Software and Applications Conference*. IEEE, 2014, pp. 384–393.

[30] M. J. Assante and R. M. Lee, "The industrial control system cyber kill chain," *SANS Institute InfoSec Reading Room*, vol. 1, 2015.

[31] O. Alexander, M. Belisle, and J. Steele, "Mitre att&ck® for industrial control systems: Design and philosophy," 2020.

[32] "Hydra source code," [Online]. Available: https://github.com/malwares/Botnet/blob/master/hydra-2008.1.zip, [Accessed February 07, 2021].

[33] P. Macgregor, "Hydra irc bot, the 25 minute overview of the kit," [Online]. Available: http://insecurety.net/hydra-irc-bot-the-25-minute-overview-of-the-kit/, [Accessed February 09, 2021].

[34] NetLab 360, "Fbot, a satori related botnet using block-chain dns system," September 2018. [Online]. Available: https://blog.netlab.360.com/threat-alert-a-new-worm-fbot-cleaning-adbminer-is-using-a- blockchain-based-dns-en/, [Accessed March 07, 2021].

[35] Genshen Ye, Hui Wang, "The new developments of the fbot," February 2019. [Online]. Available: https://blog.netlab.360.com/the-new-developments-of-the-fbot-en/, [Accessed February 02, 2023].

[36] M. Janus, "Heads of the hydra malware for network devices," 2011. [Online]. Available: https://securelist.com/heads-of-the-hydra-malware-for-network-devices/36396/, [Accessed February 09, 2021].

[37] L. Ďurfina, J. Křoustek, and P. Zemek, "Psybot malware: A step-by-step decompilation case study," in *2013 20th Working Conference on Reverse Engineering (WCRE)*, 2013, pp. 449–456.

[38] A. Remillano and M. Vicente, "Miori iot botnet delivered via thinkph exploit," December 2018. [Online]. Available: https://www.trendmicro.com/en_us/research/18/l/with-mirai-comes-miori-iot-botnet-delivered-via-thinkphp-remote- code-execution-exploit.html, [Accessed March 09, 2021].

[39] P. Čeleda, R. Krejčí, J. Vykopal, and M. Drašar, "Embedded malware-an analysis of the chuck norris botnet," in *2010 European Conference on Computer Network Defense*. IEEE, 2010, pp. 3–10.

[40] Liviu Arsene, "Hold my beer mirai – spinoff named 'liquorbot' incorporates cryptomining," Bitdefender, Tech. Rep., January 2020, [Accessed February 02, 2023]. [Online]. Available: https://www.bitdefender.com/blog/labs/hold-my-beer-mirai-spinoff-named-liquorbot-incorporates-cryptomining/

[41] "Kaiten.c source code," 2015. [Online]. Available: https://www.github.com/shipcod3/IRC-Bot-

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

Hunters/blob/master/malicious_samples/kaiten.c, [Accessed February 12, 2021].

[42] F-Secure, "Backdoor:osx/tsunami.a," [Online]. Available: https://www.f-secure.com/v-descs/backdoor_osx_tsunami_a.shtml, [Accessed February 12, 2021].

[43] Catalin Cimpanu, "New silex malware is bricking iot devices, has scary plans," ZDNet, Tech. Rep., 2019. [Online]. Available: https://www.zdnet.com/article/new-silex-malware-is-bricking-iot-devices-has-scary-plans/

[44] P. Čeleda, R. Krejčí, and V. Krmíček, "Flow-based security issue detection in building automation and control networks," in *Meeting of the European Network of Universities and Companies in Information and Communication Engineering*. Springer, 2012, pp. 64–75.

[45] "Lightaidra source code," [Online]. Available: https://github.com/ifding/iot-malware/tree/master/lightaidra, [Accessed January 22, 2023].

[46] NetLab 360, "The botnet cluster on the 185.244.25.0/24," September 2019. [Online]. Available: https://blog.netlab.360.com/the-botnet-cluster-on-185-244-25-0-24-en/, [Accessed April 14, 2021].

[47] Chao and Lei, Zhibin and Zhang, Cecilia and Hu, Aveek and Das, "Mirai variant moobot targeting d-link devices," September 2022. [Online]. Available: https://unit42.paloaltonetworks.com/moobot-d-link-devices/, [Accessed February 03, 2023].

[48] Hui and Wang, Alex and Turing, "An update for a very active ddos botnet: Moobot," July 2020. [Online]. Available: https://blog.netlab.360.com/ddos-botnet-moobot-en/, [Accessed February 03, 2023].

[49] Internet Census 2012, "Port scanning /0 using insecure embedded devices carna botnet," Internet Census 2012, Tech. Rep., 2013. [Online]. Available: http://census2012.sourceforge.net/paper.html

[50] E. Le Malécot and D. Inoue, "The carna botnet through the lens of a network telescope," in *International Symposium on Foundations and Practice of Security*. Springer, 2013, pp. 426–441.

[51] E. Kreminchuker and M. Zavodchik, "Echobot malware now up to 71 exploits, targeting scada," December 2019. [Online]. Available: https://www.f5.com/labs/articles/threat-intelligence/echobot-malware-now-up-to-71-exploits–targeting-scada, [Accessed February 01, 2023].

[52] R. Nigam, "New mirai variant adds 8 new exploits, targets additional iot devices," June 2019. [Online]. Available: https://unit42.paloaltonetworks.com/new-mirai-variant-adds-8-new-exploits-targets-additional-iot-devices/, [Accessed February 01, 2023].

[53] Broadcom, Endpoint Protection, "Linux worm targeting hidden devices," November 2013. [Online]. Available: https://community.broadcom.com/symantecenterprise/communities/community-home/librarydocuments/viewdocument?DocumentKey=6cc8a697-5c01-45ba-ad5c-599eee0a4678&CommunityKey=1ecf5f55-9545-44d6-b0f4-4e4a7f5f5e68&tab=librarydocuments, [Accessed February 12, 2021].

[54] Malware Must Die, "Mmd-0064-2019 - linux/airdropbot," September 2019. [Online]. Available: https://blog.malwaremustdie.org/2019/09/mmd-0064-2019-linuxairdropbot.html, [Accessed April 23, 2021].

[55] N. F. Tan and S. Manickam, "Wifatch: Atypical malware," in *National Advanced IPV6 Centre*. Researchgate, April 2018, doi: 10.13140/RG.2.2.27854.77123.

[56] "Wifatch source code," 2021. [Online]. Available: https://github.com/ifding/iot-malware/tree/master/linux.wifatch, [Accessed February 05, 2023].

[57] L. Tu, Y. Ma, and G. Ye, "Ttint: An iot remote access trojan spread through 2 0-day vulnerabilities," October 2020. [Online]. Available: https://blog.netlab.360.com/ttint-an-iot-remote-control-trojan-spread-through-2-0-day- vulnerabilities/, [Accessed March 18, 2021].

[58] J. B. Ullrich, "Linksys worm themoon summary: What we know so far," February 2014. [Online]. Available: https://isc.sans.edu/forums/diary/Linksys+Worm+TheMoon+Summary+What+we+know+so+far/17633, [Accessed February 13, 2021].

[59] Bitdefender Investigations and Forensics Unit, "New dark_nexus iot botnet puts others to shame," Bitdefender, Tech. Rep., 2020, [Accessed february 04, 2023]. [Online]. Available: https://www.bitdefender.com/files/News/CaseStudies/study/319/Bitdefender-PR-Whitepaper-DarkNexus-creat4349-en-EN-interactive.pdf/

[60] Aposemat Project, "Dark nexus: the old, the new and the ugly," Stratosphre Lab., Tech. Rep., June 2020, [Accessed february 04, 2023]. [Online]. Available: https://www.stratosphereips.org/blog/2020/6/8/dark-nexus-the-old-the-new-and-the-ugly

[61] J. Karasek, A. Remillano, and T. Bao, "Multistage attack delivers billgates/setag backdoor," July 2019. [Online]. Available: https://www.trendmicro.com/en_us/research/19/g/multistage-attack-delivers-billgates-setag-backdoor-can-turn- elasticsearch-databases-into-ddos-botnet-zombies.html, [Accessed February 18, 2021].

[62] M. KUZIN, "Versatile ddos trojan for linux," July 2014. [Online]. Available: https://securelist.com/versatile-ddos-trojan-for-linux/64361/, [Accessed January 22, 2023].

[63] NetLab 360, "Mozi, another botnet using dht," 2019. [Online]. Available: https://blog.netlab.360.com/mozi-another-botnet-using-dht/, [Accessed February 01, 2023].

[64] Black Lotus Labs, "New mozi malware family quietly amasses iot bots," April 2020. [Online]. Available: https://blog.lumen.com/new-mozi-malware-family-quietly-amasses-iot-bots/, [Accessed February 01, 2023].

[65] D. Fiser, J. Urbanec, and J. Horejsi, "Aesddos botnet, containers, exposed docker apis," June 2019. [Online]. Available: https://www.trendmicro.com/en_us/research/19/f/aesddos-botnet-malware-infiltrates-containers-via-exposed-docker- apis.html, [Accessed February 18, 2021].

[66] Malware Must Die, "Mmd-0026-2014 - linux/aes.ddos: Router malware warning | reversing an arm arch elf," September 2014. [Online]. Available: https://blog.malwaremustdie.org/2014/09/reversing-arm-architecture-elf-elknot.html, [Accessed January 22, 2023].

[67] Trend Micro, "Sora and unstable: 2 mirai variants target video surveillance storage systems," February 2020. [Online]. Available: https://www.trendmicro.com/vinfo/br/security/news/internet-of-things/sora-and-unstable-2-mirai-variants-target-video-surveillance-storage-systems, [Accessed February 06, 2023].

[68] D. Voolf, S. Boddy, and R. Cohen, "Gafgyt targeting huawei and asus routers and killing off rival iot botnets," December 2019. [Online]. Available: https://www.f5.com/labs/articles/threat-intelligence/gafgyt-targeting-huawei-and-asus-routers-and-killing-off-rival- iot-botnets, [Accessed February 16, 2021].

[69] T. Spring, "Bashlite family of malware infects 1 million iot devices," August 2016. [Online]. Available: https://threatpost.com/bashlite-family-of-malware-infects-1-million-iot-devices/120230/, [Accessed January 30, 2023].

[70] A. Marzano, D. Alexander, O. Fonseca, E. Fazzion, C. Hoepers, K. Steding-Jessen, M. H. Chaves, Í. Cunha, D. Guedes, and W. Meira, "The evolution of bashlite and mirai iot botnets," in *2018 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2018, pp. 00 813–00 818.

[71] "Gafgyt source code," 2021. [Online]. Available: https://github.com/ifding/iot-malware/tree/master/BASHLITE, [Accessed February 05, 2023].

[72] L. Ubiedo, "Rhombus: a new iot malware," May 2020. [Online]. Available: https://blog.apnic.net/2020/05/22/rhombus-a-new-iot-malware/, [Accessed April 23, 2021].

[73] Malware Must Die, "Mmd-0028-2014 - linux/xor.ddos : Fuzzy reversing a new china elf," September 2014. [Online]. Available: https://blog.malwaremustdie.org/2014/09/mmd-0028-2014-fuzzy-reversing-new-china.html, [Accessed February 19, 2021].

[74] P. Kalnai and J. Horejsi, "Ddos trojan: A malicious concept that conquered the elf format," in *Proc. Virus Bull. Conf*, 2015, pp. 62–70.

[75] Joe Sandbox, "Analysis report hoho.x86," [Online]. Available: https://www.joesandbox.com/analysis/207020/0/lighthtml, [Accessed April 07, 2021].

[76] Malware Must Die, "Mmd-0030-2015 - new elf malware on shellshock: the chinaz," January 2015. [Online]. Available: https://blog.malwaremustdie.org/2015/01/mmd-0030-2015-new-elf-malware-on.html, [Accessed February 22, 2021].

[77] K. Hsu, Z. Zhang, and R. Nigam, "New mirai variant targets zyxel network-attached storage devices," Palo Alto Networks, Unit 42, Tech. Rep., March 2020. [Online]. Available: https://unit42.paloaltonetworks.com/new-mirai-variant-mukashi///

[78] Greg Belding, "Mukashi malware: What it is, how it works and how to prevent it malware spotlight," June 2020. [Online]. Available: https://resources.infosecinstitute.com/topic/mukashi-malware-what-it-is-how-it-works-and-how-to-prevent-it-malware-spotlight/, [Accessed February 06, 2023].

[79] O. Bilodeau and T. Dupuy, "Dissecting linux/moose," We Live Security by ESET, Tech. Rep., May

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

2015. [Online]. Available: https://www.welivesecurity.com/wp-content/uploads/2015/05/Dissecting-LinuxMoose.pdf

[80] NetLab 360, "The gafgyt variant vbot seen in its 31 campaigns," July 2020. [Online]. Available: https://blog.netlab.360.com/the-gafgyt-variant-vbot-and-its-31-campaigns/, [Accessed March 18, 2021].

[81] Malware Must Die, "Mmd-0055-2016 - linux/pnscan ; elf worm that still circles around," August 2016. [Online]. Available: https://blog.malwaremustdie.org/2016/08/mmd-0054-2016-pnscan-elf-worm-that.html, [Accessed February 17, 2021].

[82] "Pnscan source code," 2021. [Online]. Available: https://github.com/ifding/iot-malware/tree/master/pnscan, [Accessed February 05, 2023].

[83] NetLab 360, "Kaiji: New chinese linux malware turning to golang," May 2020. [Online]. Available: https://www.intezer.com/blog/research/kaiji-new-chinese-linux-malware-turning-to-golang/, [Accessed April 04, 2021].

[84] M. Malik and M.-E. M.Léveillé, "Meet remaiten – a linux bot on steroids targeting routers and potentially other iot devices," We Live Security by ESET, Tech. Rep., March 2016. [Online]. Available: https://www.welivesecurity.com/2016/03/30/meet-remaiten-a-linux-bot-on-steroids-targeting-routers-and- potentially-other-iot-devices/

[85] Avira Protection Labs, "Katana: a new variant of the mirai botnet," October 2020. [Online]. Available: https://prod-blog.avira.com/katana-a-new-variant-of-the-mirai-botnet, [Accessed April 26, 2021].

[86] C. Xiao, C. Zheng, and Y. Jia, "New iot/linux malware targets dvrs, forms botnet," *Palo Alto Networks, Unit 42*, vol. 9, 2017.

[87] NetLab 360, "Heh, a new iot p2p botnet going after weak telnet services," October 2020. [Online]. Available: https://blog.netlab.360.com/heh-an-iot-p2p-botnet/, [Accessed April 26, 2021].

[88] Malware Must Die, "Mmd-0058-2016 - linux/nyadrop - a linux mips iot bad news," October 2016. [Online]. Available: https://blog.malwaremustdie.org/2016/10/mmd-0058-2016-elf-linuxnyadrop.html, [Accessed February 21, 2021].

[89] NetLab 360, "New threat: Zhtrap botnet analysis report," March 2021. [Online]. Available: https://blog.netlab.360.com/new_threat_zhtrap_botnet_cn/, [Accessed February 06, 2023].

[90] Alex Turing, "Zhtrap botnet uses honeypot to harvest infected devices," May 2021. [Online]. Available: https://blog.apnic.net/2021/05/04/zhtrap-botnet-uses-honeypot-to-harvest-infected-devices/, [Accessed February 06, 2023].

[91] CyberX, "Radiation iot cyber security campaign," CyberX by Micrrosoft, Tech. Rep., 2016. [Online]. Available: https://cdn2.hubspot.net/hubfs/2479124/CyberX_-_Campaign_Radiation-1.pdf?__hstc=6343789.7982866e313d52ac6230a09a2ff95a0e.1569848719445.1569848719445.1569848719445.1&__hssc=6343789.1.1569848719446&t=1475330621687&submissionGuid=144b86c3-270a-4ead-aa6f-61e0d4eb5d09

[92] Malware Must Die, "Mmd-0057-2016 - linux/luabot - iot botnet as service," September 2016. [Online]. Available: https://blog.malwaremustdie.org/2016/09/mmd-0057-2016-new-elf-botnet-linuxluabot.html#ddos, [Accessed February 25, 2021].

[93] Radware Threat Alerts, "Dark.iot botnet," August 2021. [Online]. Available: https://www.radware.com/security/threat-advisories-and-attack-reports/dark-iot-botnet/, [Accessed February 06, 2023].

[94] Vaibhav and Singhal, Ruchna and Nigam, Zhibin and Zhang, Asher and Davila, "New mirai variant targeting network security devices," March 2021. [Online]. Available: https://unit42.paloaltonetworks.com/mirai-variant-iot-vulnerabilities/, [Accessed February 06, 2023].

[95] H. Sinanović and S. Mrdovic, "Analysis of mirai malicious software," in *2017 25th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*. IEEE, 2017, pp. 1–5.

[96] A. Dulaunoy, G. Wagener, S. Mokaddem, and C. Wagner, "An extended analysis of an iot malware from a blackhole network," in *Proceedings of the Networking Conference TNC*, vol. 17, 2017.

[97] "Mirai source code," 2017. [Online]. Available: https://github.com/jgamblin/Mirai-Source-Code, [Accessed February 05, 2023].

[98] Nozomi Networks, "Threat intelligence: Analysis of the sbidiot iot malware," April 2021. [Online]. Available: https://www.nozominetworks.com/blog/threat-intelligence-analysis-of-the-sbidiot-iot-malware/, [Accessed February 06, 2023].

[99] Jie Jİ, "Analysis of the sbidiot iot malware," May 2021. [Online]. Available: https://nsfocusglobal.com/analysis-of-the-sbidiot-iot-malware/, [Accessed February 06, 2023].

[100] S. Edwards and I. Profetis, "Hajime: Analysis of a decentralized internet worm for iot devices," *Rapidity Networks*, vol. 16, pp. 1–18, 2016.

[101] S. Herwig, K. Harvey, G. Hughey, R. Roberts, and D. Levin, "Measurement and analysis of hajime, a peer-to-peer iot botnet," in *Network and Distributed Systems Security (NDSS) Symposium*, 2019.

[102] Lacework Labs, "Mirai goes stealth – tls & iot malware," September 2021. [Online]. Available: https://www.lacework.com/fr/blog/mirai-goes-stealth-tls-iot-malware/, [Accessed September 16, 2022].

[103] Malware Must Die, "Mmd-0059-2016 - linux/ircTelnet (new aidra) - a ddos botnet aims iot w/ ipv6 ready," October 2016. [Online]. Available: https://blog.malwaremustdie.org/2016/10/mmd-0059-2016-linuxirctelnet-new-ddos.html, [Accessed February 28, 2021].

[104] Lisa Vaas, "Botenago botnet code leaked to github, impacting millions of devices," January 2022. [Online]. Available: https://threatpost.com/botenago-botnet-code-leaked-to-github/178059/, [Accessed September 16, 2022].

[105] Trend Micro, "Brickerbot malware emerges, permanently bricks iot devices," April 2017. [Online]. Available: https://www.trendmicro.com/vinfo/us/security/news/internet-of-things/brickerbot-malware-permanently-bricks-iot-devices, [Accessed February 28, 2021].

[106] Joie Salvio, "Manga aka dark mirai-based campaign targets new tp-link router rce vulnerability," December 2021. [Online]. Available: https://www.fortinet.com/blog/threat-research/manga-aka-dark-mirai-based-campaign-targets-new-tp-link-router-rce-vulnerability, [Accessed October 08, 2022].

[107] T. Yeh, D. Chiu, and K. Lu, "Persirai: New iot botnet targets ip cameras," May 2017. [Online]. Available: https://www.trendmicro.com/en_us/research/17/e/persirai-new-internet-things-iot-botnet-targets-ip-cameras.html?_ga=2.116930950.1850548625.1614865704-1937216687.1614209380, [Accessed February 28, 2021].

[108] Joie Salvio and Roy Tay, "Fresh totolink vulnerabilities picked up by beastmode mirai campaign," April 2022. [Online]. Available: https://www.fortinet.com/blog/threat-research/totolink-vulnerabilities-beastmode-mirai-campaign, [Accessed February 06, 2023].

[109] Dr. Web Anti-Virus, "Linux.muldrop.14," June 2017. [Online]. Available: https://vms.drweb.com/virus/?_is=1&i=15389228, [Accessed February 28, 2021].

[110] Joie Salvio and Roy Tay, "Enemybot: A look into keksec's latest ddos botnet," April 2022. [Online]. Available: https://www.fortinet.com/blog/threat-research/enemybot-a-look-into-keksecs-latest-ddos-botnet, [Accessed October 08, 2022].

[111] Oleg and Kolesnikov, Den and Iuzvyk, Tim and Peck, "Detecting enemybot – securonix initial coverage advisory," May 2022. [Online]. Available: hhttps://www.securonix.com/blog/detecting-the-enemybot-botnet-advisory, [Accessed February 06, 2022].

[112] Genshen Ye, "Iot_reaper: A rappid spreading new iot botnet," October 2017. [Online]. Available: https://blog.netlab.360.com/iot_reaper-a-rappid-spreading-new-iot-botnet-en/, [Accessed February 01, 2023].

[113] ——, "Iot_reaper: A few updates," October 2017. [Online]. Available: https://blog.netlab.360.com/iot_reaper-a-few-updates-en/, [Accessed February 01, 2023].

[114] Nozomi Networks Labs, "New botenago variant discovered by nozomi networks labs," April 2022. [Online]. Available: https://www.nozominetworks.com/blog/new-botenago-variant-discovered-by-nozomi-networks-labs/, [Accessed October 15, 2022].

[115] Trend Micro, "Fbot aka satori is back with new peculiar obfuscation, brute-force techniques," December 2019. [Online]. Available: https://www.trendmicro.com/vinfo/us/security/news/internet-of-things/fbot-aka-satori-is-back-with-new-peculiar-obfuscation-brute-force-techniques, [Accessed March 02, 2021].

[116] Radware, "Satori iot botnet variant," June 2018. [Online]. Available: https://www.radware.com/security/ddos-threats-attacks/threat-advisories-attack-reports/satori-iot-botnet/, [Accessed February 02, 2022].

[117] Joie Salvio and Roy Tay, "So rapperbot, what ya bruting for?" August 2022. [Online]. Available: https://www.fortinet.com/blog/threat-research/rapperbot-malware-discovery, [Accessed February 07, 2023].

[118] Bogdan Botezatu, "New hide 'n seek iot botnet using custom-built peer-to-peer communication spotted in the wild," Bitdefender, Tech. Rep.,

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3262614

2018. [Online]. Available: https://labs.bitdefender.com/2018/01/new-hide-n-seek-iot-botnet-using-custom-built-peer-to-peer- communication-spotted-in-the-wild/

[119] Black Lotus Labs, "Zuorat hijacks soho routers to silently stalk networks," June 2022. [Online]. Available: https://blog.lumen.com/zuorat-hijacks-soho-routers-to-silently-stalk-networks/, [Accessed October 29, 2022].

[120] J. C. Sapalo Sicato, P. K. Sharma, V. Loia, and J. H. Park, "Vpnfilter malware analysis on cyber threat in smart home network," *Applied Sciences*, vol. 9, no. 13, p. 2763, 2019.

[121] William Largent, "New vpnfilter malware targets at least 500k networking devices worldwide," May 2018. [Online]. Available: https://blog.talosintelligence.com/vpnfilter/, [Accessed February 02, 2023].

[122] DAN GOODIN, "Shikitega — new linux malware combines unusual stealth with a full suite of capabilities," September 2022. [Online]. Available: https://arstechnica.com/information-technology/2022/09/new-linux-malware-combines-unusual-stealth-with-a-full-suite-of-capabilities/, [Accessed October 29, 2022].

[123] Minh Tran , "Shinoa, owari, mirai: What's with all the anime references?" May 2018 2019. [Online]. Available: https://www.fortinet.com/blog/threat-research/shinoa–owari–mirai–what-s-with-all-the-anime-references-, [Accessed February 02, 2023].

[124] Black Lotus Labs, "Chaos is a go-based swiss army knife of malware," September 2022. [Online]. Available: https://blog.lumen.com/chaos-is-a-go-based-swiss-army-knife-of-malware/, [Accessed October 29, 2022].

[125] K. Angrishi, "Turning internet of things (iot) into internet of vulnerabilities (iov): Iot botnets," *arXiv preprint arXiv:1702.03681*, 2017.

[126] M. De Donno, N. Dragoni, A. Giaretta, and A. Spognardi, "Ddos-capable iot malwares: Comparative analysis and mirai investigation," *Security and Communication Networks*, vol. 2018, 2018.

[127] Q.-D. Ngo, H.-T. Nguyen, L.-C. Nguyen, and D.-H. Nguyen, "A survey of iot malware and detection methods based on static features," *ICT Express*, 2020.

[128] E. Cozzi, P.-A. Vervier, M. Dell'Amico, Y. Shen, L. Bilge, and D. Balzarotti, "The tangled genealogy of iot malware," in *Annual Computer Security Applications Conference*, 2020, pp. 1–16.

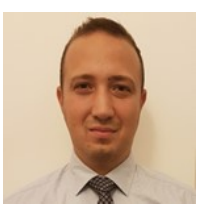
[129] D.-W. Kim, G.-Y. Shin, and M.-M. Han, "Analysis of feature importance and interpretation for malware classification," *Computers, Materials & Continua*, vol. 65, no. 3, pp. 1891–1904, 2020.

[130] Bill Toulas, "Beastmode botnet boosts ddos power with new router exploits," April 2022. [Online]. Available: hhttps://www.bleepingcomputer.com/news/security/beastmode-botnet-boosts-ddos-power-with-new-router-exploits/, [Accessed February 06, 2023].

[131] Joie Salvio and Roy Tay, "New rapperbot campaign – we know what you bruting for this time," November 2022. [Online]. Available: https://www.fortinet.com/blog/threat-research/new-rapperbot-campaign-ddos-attacks, [Accessed February 07, 2023].

[132] Fernando and Merces, Augusto and Remillano II, Jemimah and Molin, "Mirai botnet attack iot devices via cve-2020-5902," July 2020. [Online]. Available: https://www.trendmicro.com/en_us/research/20/g/mirai-botnet-attack-iot-devices-via-cve-2020-5902.html, [Accessed February 06, 2023].

[133] S. Hilt, R. McArdle, F. Merces, M. Rosario, , and D. Sancho, "Uncleanable and unkillable: The evolution of iot botnets through p2p networking," Trend Micro Research, Tech. Rep., March 2021.



IBRAHIM GULATAS received the B.S. degree in Computer Engineering from Turkish Naval Academy, Turkey, in 2010, and the M.S. degree in Computer Engineering from Bahcesehir University, Turkey, in 2018. Currently, he is a Ph.D. Candidate in Istanbul Commerce University, Turkey. His current research interests include information security and malware analysis.



H. HAKAN KILINC is working as an Innovation Project Manager for Orion Innovation Turkey since December 2019. He worked as a cybersecurity product line manager for Netas between 2014 and 2019. He received his B.S. degree in Mathematics and Computer Science from Aegean University, Izmir, Turkey, in 1997. He received his M.S. degree in 2001 in the Department of Computer Engineering at the Izmir Institute of Technology. He worked as a visiting scholar at the University of Texas at Dallas between 2009 and 2011. He holds a Ph.D. about the security of SIP (Session Initiation Protocol) from the Department of Electronics Engineering at the Gebze Technical University in 2014.



A. HALIM ZAIM is currently a faculty member in the Department of Computer Engineering at the Faculty of Engineering at Istanbul Commerce University, and the Director of the Center for Information Technology Application and Research at Istanbul Commerce University. Abdul Halim Zaim has been serving as Vice-Rector and Director of the Technology Transfer Office. His research interest mainly focuses on IoT, Big Data, Network Design, Cyber Security, Network Security, and Communication- Network Protocols. He received his MS degree in Computer Engineering from Bogazici University in 1996 and his Ph.D. in Electrical and Computer Engineering from North Carolina State University (NCSU) in 2001.



M. ALI AYDIN is currently working as an associate professor at the Computer Engineering Department of Istanbul University-Cerrahpasia. His interest mainly focuses on Cyber Security, Cryptography, Network Security, and Communication-Network Protocols. He received his Ph.D. degree in Computer Engineering from Istanbul University and he has completed Post Doctoral research at Telecom SudParis in the Department of Computer Science.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4