*future internet*

MDPI

*Article*

# A Survey on Botnets: Incentives, Evolution, Detection and Current Trends

**Simon Nam Thanh Vu †, Mads Stege †, Peter Issam El-Habr †, Jesper Bang † and Nicola Dragoni *,‡**

DTU Compute, Technical University of Denmark, 2800 Kgs. Lyngby, Denmark;
s200361@student.dtu.dk (S.N.T.V.); s165243@student.dtu.dk (M.S.); s165202@student.dtu.dk (P.I.E.-H.);
s144211@student.dtu.dk (J.B.)
* Correspondence: ndra@dtu.dk; Tel.: +45-45-25-37-31
† These authors contributed equally to this work.
‡ Current address: Richard Petersens Plads, 2800 Kgs. Lyngby, Denmark.

**Abstract:** Botnets, groups of malware-infected hosts controlled by malicious actors, have gained prominence in an era of pervasive computing and the Internet of Things. Botnets have shown a capacity to perform substantial damage through distributed denial-of-service attacks, information theft, spam and malware propagation. In this paper, a systematic literature review on botnets is presented to the reader in order to obtain an understanding of the incentives, evolution, detection, mitigation and current trends within the field of botnet research in pervasive computing. The literature review focuses particularly on the topic of botnet detection and the proposed solutions to mitigate the threat of botnets in system security. Botnet detection and mitigation mechanisms are categorised and briefly described to allow for an easy overview of the many proposed solutions. The paper also summarises the findings to identify current challenges and trends within research to help identify improvements for further botnet mitigation research.

**Keywords:** botnet; malware; security; IoT

check for updates

**Citation:** Thanh Vu, S.N.; Stege, M.; El-Habr, P.I.; Bang, J.; Dragoni, N. A Survey on Botnets: Incentives, Evolution, Detection and Current Trends. *Future Internet* **2021**, *13*, 198. https://doi.org/10.3390/fi13080198

Academic Editors: Ammar Alazab, Mamoun Alazab, Ansam Khraisat and Savitri Bevinakoppa

Received: 17 June 2021
Accepted: 29 July 2021
Published: 31 July 2021

## 1. Introduction

Botnets are one of the most prominent threats to system and IoT security in the recent age of cloud-enabled pervasive computing. New pervasive computing architectures, such as always-on mobile devices and Internet-of-Things, provide additional infection vectors for botnet attacks. Due to the large increase in interconnected devices and system platforms, the types and attack patterns of botnets are constantly changing [1–3]. As an example, the IoT botnet Mirai has seen growth from approx. 143,000 occurrences to 225,000 occurrences from 2018 to 2019 alone [4]. For these reasons, it is important to first get an understanding of the anatomy of botnets, their evolution up until now and what mitigation mechanisms and tools are available to combat botnet-based attacks.

A botnet is a network of malware-infected hosts, which are typically controlled by a Command and Control (C&C) server. The C&C server architecture allows for distributed malicious attacks on either the infected hosts or other interconnected hosts over LAN or the internet [5,6]. C&C servers are commonly known as the *botmasters*, while infected hosts are simply referred to as *bots* [1].

Botnets are commonly divided into two general architectural structures, centralised and Peer-to-Peer (P2P). These structures are defined by how commands are transmitted throughout the C&C channel. In centralised botnets, as seen in Figure 1, a central C&C server is responsible for sending commands to bots. Meanwhile, in a P2P network, the botnet commands are propagated throughout the P2P overlay network, as seen in Figure 2.

Centralised botnets are usually more efficient but are less resilient to countermeasures, as the centralised C&C server acts as a single point of failure for the entire botnet [6,7].

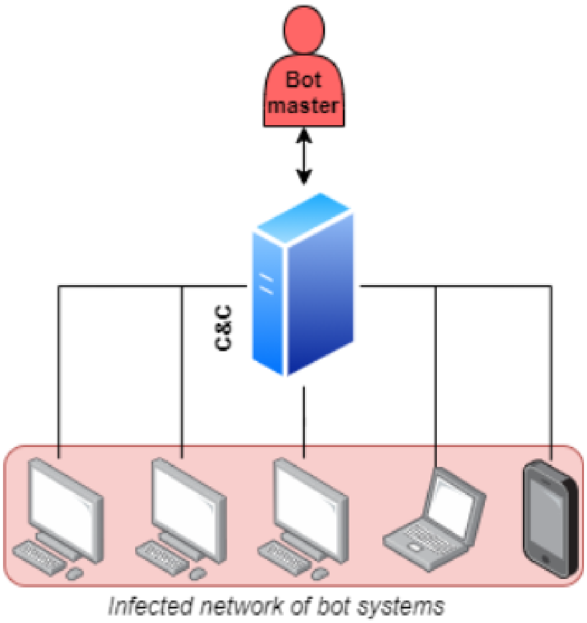

**Figure 1.** Example of a centralised C&C botnet structure.

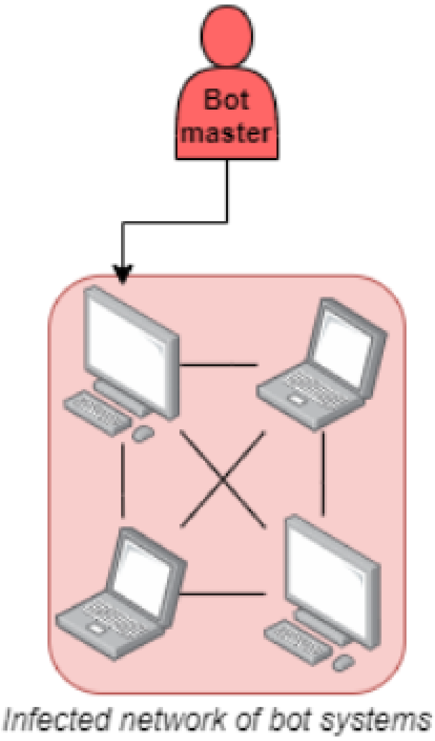

**Figure 2.** Example of a decentralised (P2P) C&C botnet structure.

Botnets can be used for numerous kinds of distributed attacks such as Distributed Denial of Service (DDoS) attacks, malicious software distribution, piracy, extortion and many others. Initially, botnets spread by the use of Internet Relay Chat (IRC), but presently, the attack vectors of botnets are much more varied. These attack vectors include file-sharing networks, infected email attachments, infected websites and vulnerability attacks [1,8]. The rise of internet-connected pervasive devices provides botnets with a larger attack surface and more vulnerable hosts to infect. Prominent botnet attacks such as Mirai and Zeus show how the pervasive era of computing and the interconnected internet has caused the rise and evolution of increasingly complex botnets, making continued research within the field pertinent [2,9,10].

### *1.1. Contribution and Research Questions*

This systematic literature review presents a survey on the incentives and evolution of botnets as well as detection and mitigation mechanisms developed to combat botnets. The main contribution of this paper is a diverse overview of these topics according to mostly peer-reviewed literature during the period 2005–2021, with a particular focus on botnet detection and mitigation. The second contribution is an analysis of the evolution of botnets and mitigation strategies in order to develop an idea of the current trends and challenges within the field of botnets.

The specific research questions asked by this paper are:

1. What incentives are behind the development of botnet attacks?
2. How have botnet attacks evolved over time?
3. What has the research industry proposed to mitigate the threat of botnets?
4. What current trends and challenges related to botnets have been identified by contemporary research?

### *1.2. Outline*

The paper is laid out as a systematic literature review with particular focus on botnet detection and corresponding mitigation mechanisms to identify current trends in botnet attacks. Section 1 gives a general introduction to botnets, as well as the research questions of this paper. Section 2 describes the previous surveys and literature reviews made by other researchers to describe the potential contribution of this paper. Section 3 describes the methodology used for the paper. Sections 4 and 5 cover the incentives and evolution of botnet attacks respectively, giving an overview of the development and reasoning behind this kind of attack (research questions 1 and 2). Section 6 details the different mitigation and detection mechanisms proposed in research to combat botnets (research question 3). Section 7 provides an analysis on the development and trends in botnets and how to potentially mitigate current attacks (research question 4). Lastly, Section 8 concludes the paper.

## 2. Related Work

Many surveys and systematic literature reviews on botnets can be found in the literature, although their scope and focus vary significantly. Table 1 gives an overview of such related works, with emphasis on their main contribution and on how this paper can enhance the state of the art on botnets research.

**Table 1.** Novelty of this paper with respect to related surveys. Number of references within each research question and year range is compared with the contribution of this paper to quantitatively show the novelty of this paper (years: 2006–2021, incentives: 13 references, evolution: 33, detection/mitigation: 134, trends/challenges: 41). For rows with multiple references, a shorthand format (*ref—numOfPapers, yearSpanOfReferences)* is used.

| Paper | Main Contribution and Reference Metrics | This Paper |
|---|---|---|
| [11] | Offers only generalised information about botnets and botnet detection/mitigation strategies. Thirty-nine references from 2007 to 2012. | Describes specific botnet detection mechanisms, advantages, disadvantages, for instance, the *Shieldnet* framework to detect botnets in vehicular networks [12]. |
| [13,14] | Focuses on describing different kinds of botnet attacks ([13]) and on the threats represented by botnets ([14]), without going into specific mitigation strategies. ([13]—27 references from 2006 to 2016) ([14]—60 references from 2003 to 2018) | Describes both botnet evolution and threats, and offers insight into different detection and mitigation mechanisms. |

**Table 1.** *Cont.*

| Paper | Main Contribution and Reference Metrics | This Paper |
|---|---|---|
| [1,15–17] | Offer great insight on botnet research, types of botnets as well as detection and mitigation mechanisms. However, both papers do not include more recent studies and publications (all are pre-2014). ([1]—205 references), ([7]—36), ([15]—49), ([16]—217), ([17]—28). | Covers the same points as aforementioned papers, but also includes more recent research from 2014–2021 such as [18,19] and more. |
| [20] | Presents potential challenges of mobile botnets, but does not include any more recent research (paper from 2012). 40 references from 2002–2012. | Presents more recent papers on mobile botnets such as [18,19] and more. |
| [7,21]. | Covers only generalised botnet types and few specific recent types, such as cloud botnets and social botnets ([7]) or covers P2P botnets only ([21]). ([7]—25 references from 2015–2015) ([7]—36, 2005–2013) | Covers more kinds of botnet types such as IoT botnets, mobile botnets, VANET-based botnets and their related challenges and trends. |
| [10] | Describes botnet evolution, attack threats and actors, but not go into detection and mitigation techniques against botnets. Thirty-one references from 1998–2009. | Covers the same points and also describes different detection categories and specific mitigation mechanisms. |
| [22–27] | Limited scope of botnet detection techniques ([22]—34 detection/mitigation references from 1997–2008), ([23]—38, 2004–2011), ([24]—9, 2008–2019), ([25]—7, 2019–2017), ([26]—11, 2008–2015), ([27]—20, 2004–2013). | Includes a larger breadth of more recent detection papers, such as [28,29] and more than 100 more papers compared to [22]. |
| [30] | Focuses specifically on DDoS botnet attacks without covering detection strategies. 145 references from 1993–2015. | Describes potential attack threats of botnets while also covering detection mechanisms. |
| [31,32] | Mentions only IoT-based botnets. ([31]—36 references from 2010 to 2019), ([32]—122, 2004–2021). | Covers IoT-based botnets and also includes other types of botnets, such as mobile botnets, social botnets and VANET-based botnets. |
| [33,34] | Discusses various botnet detection categories in general, but does not highlight the specifics of each technique. ([33]—34 references from 2005 to 2010). | Highlights and describes each detection technique individually including the strengths and novelty of each botnet detection approach. |
| [35,36] | Only compares machine-learning based botnet detection techniques. ([35]—38 references from 1995–2020), ([36]—25, 2001–2017) | Compares machine-learning based detection techniques as well as many more types (IoT, social botnets and more). |

Reference [11] from 2012 gives a short overview of botnets characteristics, their activities, detection mechanisms and challenges. The survey is, with 39 references, quite limited in scope. Likewise, papers such as [13,14] provide pertinent introductions to the topics covered by the research questions of this paper. Like [11], however, the papers do not quite cover the breadth and depth of available botnet research however.

Reference [1] is an excellent literature review on botnets and goes into more depth on the general topic of botnets with a detailed timeline of botnets from 1993 and beyond. The literature review also goes into defence mechanisms, the then-current scope of detection techniques and future challenges. The paper is a bit older (2013) and therefore lacks some of the newer developments in botnet detection and mitigation. Likewise, Reference [16] also touches upon the topics of detection, mitigation, future challenges and evolution, but is also a bit on the older side (2014). Reference [7] also discusses the current challenges, defence mechanisms and suggested mitigation techniques. The paper, however, limits the scope of these discussions to purely P2P-based botnets. Reference [20] investigates botnets on mobile devices and their potential damage, but is limited by its age and the relative newness of smartphone technology at the time (2012).

Other surveys and detection comparisons, such as [17,21,24–27,33–36], also focus on detection and mitigation mechanisms. Common among them is that they primarily focus on detection techniques and comparing the effectiveness of the techniques in limited scenarios. This is a factor which this paper attempts to remedy, by also including mitigation mechanisms as well as adding a more broad perspective on botnets in general.

Some earlier papers such as [10] discuss the threats botnets pose to the general information security landscape. The paper looks into how law enforcement can act upon the criminals behind botnets and focuses mostly on botnets from the perspective of information security. Reference [10] does not, however, go into specific detection or mitigitation mechanisms. Reference [22] touches upon and analyses the use of honeynets, honeypots, signature-based detection with IDS, anomaly-based detection with network analysers such as Botsniffer, mining-based detection and DNS-based detection. Furthermore, the survey explores the use of abnormally recurring NXDOMAIN reply rates as a method of detection. The survey is quite limited in scope; for instance, the paper only presents 13 different papers within botnet detection approaches while this paper has more than 100. Reference [23] proposes detection, prevention, investigation and mitigation techniques by classifying the evolved strategies into five categories: anomaly, signature, DNS, data mining and hybrid technique. Again, the paper is limited in scope with only 39 different detection papers mentioned. Reference [25] addresses four different major botnet detection approaches: signature-based, anomaly-based, DNS-based and mining based detection but does so with four pages and only seen papers mentioned. Reference [30] focuses primarily on botnets used in DDoS attacks. The paper goes into depth about the life cycle, communication mechanisms and attack types within DDoS-enabled botnets. The paper does not discuss any mitigation mechanisms, however. Reference [31] is another survey with a specific focus, namely IoT botnets, which gives a very good introduction to the specific topic of IoT botnets, but otherwise does not cover any other kinds of botnets. Reference [24] endorses convolutional neural network (CNN) as being one of the best-performing techniques for detecting botnets in IoT devices. While a newer systematic review [32] answers the questions of how IoT botnets are formed, what kind of communication and scenarios involve IoT botnets, and which methods currently exist to detect IoT botnets.

A detailed survey [15] touches on problems with other botnet detection papers such as the lack of public dataset, lack of comparison with other papers, very few botnets in datasets, inaccurate outcomes of experiments and more. According to [15], the general best practice of botnet detection is using the most general behavioural features to generate a hybrid detection method where multiple detection algorithms work together as botnets evolve faster than ever. Furthermore, the paper appeals for dataset improvements and studies to compare methods used in detection. Like [1], the paper is a bit on the older side (2013).

While many surveys have gone into great depth on specific areas of botnets, such as detection, it is the opinion of the authors of this paper that a comprehensive systematic literature review with updated literature is needed. Like [1,15], the paper should focus on the current state of botnet evolution, detection, mitigation and current trends and challenges, as well as provide new insights and ideas through more recent (2013+) research. This will allow the research community a more holistic source of reference for the current state of botnets in 2021.

## 3. Methodology

This section describes the search and paper selection methodology used to select literature for this paper. The methodology contains elements from both [37,38], which provide guidelines on how to write a systematic literature review and how to use snowball sampling for paper inclusion respectively. An overview of each step of the paper selection process can be found in Figure 3, with more detailed description of each step being described later.

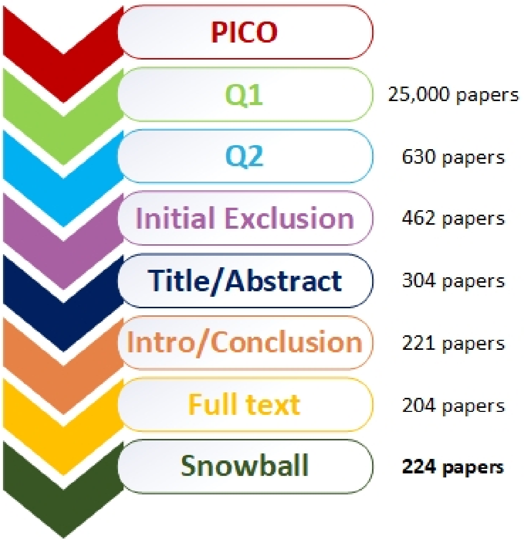

**Figure 3.** Methodology steps for paper selection and how many papers were left at each exclusion/inclusion step.

### *3.1. Search Strategy*

The PICO (Population, Intervention, Comparison and Outcomes) criteria to identify relevant search queries from the paper's research questions [39]. The criteria for this specific paper are defined as follows:

- *Population:* The paper is interested in all research focused on botnet incentives, evolution, detection and mitigation, including other surveys. Malware in general is considered too broad, and only papers focused specifically on botnets are included.
- *Intervention:* Does not apply as all papers within the research space of botnets are interesting for the purpose of the survey.
- *Comparison:* Different approaches to the detection of botnets in particular are compared to identify advantages/disadvantages. The frequency distribution of detection and mitigation mechanisms described in papers are also compared.
- *Outcomes:* Expected results are an overview of botnet progression and mitigation mechanisms as well as an identification of current trends based on the aforementioned overview.

Two important keywords were identified from these criteria, *Botnet* and *Security*. As the main goal of the paper is to provide a mitigation-oriented analysis of botnet papers and current trends in botnet security, these two keywords were deemed as the most important.

Initially, five sources—Google Scholar, DTU FindIt, ACM Digital Library, Scopus and IEEE Explore—were used for database query of botnet papers. The first query of ***'Botnet'* in title** produced more than 18,200 papers, too much to realistically process. Additionally, IEEE Explore, DTU FindIt, ACM Digital Library and Scopus found 1455, 5912, 1379 and 3411 papers respectively. Instead, a second query: ***'botnet'* in the title and *'security'* in abstract** was used to both exclude some potentially unrelated papers and to include both the identified keywords.

For the second query, Google Scholar was removed as it did not provide the option of searching within abstracts. In total, 306, 224, 85 and 399 papers were found on DTU FindIt, IEEE Explore, ACM Digital Library and Scopus, respectively, with the new query. This query did find multiple duplicates between the sources that were removed in the first exclusion step. In total, some ~630 papers in total (unique, not including duplicates) were found.

### *3.2. Exclusion/Inclusion Process*

Several exclusion steps and one inclusion step were executed to identify which papers to include in this paper.

#### 3.2.1. Initial Exclusion

The initial exclusion step excluded papers based on the following exclusion criteria, any papers not meeting the all criteria were excluded

- Papers from 2005 or newer
- English language papers only
- Botnet-related papers only
- Open Access or free for DTU students to read through DTU FindIt
- Has a Digital Object Identifier (DOI) [40]
- Peer Reviewed.

Excluding papers older than 2005 might mean some of the initial papers on botnets might be missed. However, because backwards snowball sampling of references is used later, those papers should be included during that step. Only two papers were excluded because they were not available through DTU FindIt due to a paywall. The remaining number of papers was 462 at this phase. Some non-peer-reviewed internet sources were included in the paper for definitions or additional perspectives.

#### 3.2.2. Title and Abstract Review

After the initial exclusion each paper was assigned to one reviewer for a quick title and abstract review. The purpose of this exclusion step was twofold: first to exclude any irrelevant papers and second to identify which research questions could be answered by the paper (e.g., other survey, detection paper, mitigation). If the abstract of a paper did not give any indication of being useful for the research questions, the paper was excluded. A total of 304 papers were included in the next step.

#### 3.2.3. Introduction/Conclusions Review

The penultimate exclusion step involved a review of title, abstract, introduction and conclusion of each paper, with two reviewers being assigned to each paper. Reviewers were assigned to papers that they did not review in the previous exclusion step, allowing for a total of three different reviewer opinions on all papers. Each paper was excluded if one reviewer found the paper either lacking or otherwise irrelevant for this paper. This step was also used to classify the contents of each paper in subcategories, e.g., detection papers focusing on machine learning approaches or detection papers focusing on API call logs. The writing of each paper was also considered. A paper was excluded if both reviewers had issues understanding the main purpose of the paper. A total of 221 papers were left after this review.

#### 3.2.4. Full Text Review

A final full text review was performed for the remaining papers. Reviewers were reassigned the papers they reviewed for the previous step to exclude any redundant papers. A short summary for each included paper was written in order to allow all reviewers to understand the contribution of each paper, without reading it themselves. At this point 204 papers remained with a certain guarantee of being useful for the purpose of this literature review.

#### 3.2.5. Backwards Snowball Sampling

Finally, a backwards snowball sampling method was used to include any papers that were missed during the initial query. The process involved going through the references of each included paper and see if any reference might be relevant for the purpose of this paper. After snowballing, the final number of peer-reviewed references included in this paper was 224.

## 4. Incentives

For the purpose of clarification, Table 2 below details a number of papers discussed in this section.

As to the purpose and incentive of botnets, a great many differing desires may be present. This is in no small part due to the multitude of different targets and aspirations for the various botnets. To further complicate matters, not all botnets are necessarily entirely malicious. There exist both malevolent and benevolent botnets, seeking out potential targets to further their respective inherent agendas. The latter of these will be touched on in Section 4.2. For now, the malevolent type of botnets will be the focus of attention.

**Table 2.** Table of motivations behind botnet-based attacks. The columns describe the motivation, type of attack, known affected targets, attack vector(s) and the case study/paper describing the attack.

| Motivation | Type of Attack | Target | Vector of Attack | Papers |
|---|---|---|---|---|
| Disruption<br>Denial of service. | DDoS. | Hosting Service Provider. | IoT devices. | [41] |
| Political affiliation<br>Censorship. | DDoS. | Military complex computers. | C&C-based botnet. | [42] |
| Disruption and Destabilisation of national power grids. | DDoS. | Metering infrastructure. | Internet-connected computers and IoT devices. | [43] |
| Sensitive data<br>Password cracking. | Cracking/Brute-forcing. | Common people. | Various. | [44–46] |
| Cyber espionage. | (Spear-)phishing and malware | Multiple victims listed. | Various. | [31,47] |
| Security patching<br>Vulnerability scanning (benevolent). | Security patching | Unsecure IoT devices. | Other IoT devices. | [48] |
| Botnet spoofing (fighting other botnets). | Mitigation. | Malicious botnets. | Existing botnets. | [49] |
| Miscellaneous pieces of work detailing motivations. | Various types discussed. | Various targets discussed. | Multiple types discussed. | [10,50] |

### *4.1. Malevolent Botnets*

In the world of malevolent botnets, there exist two main types of incentives for the development of a botnet. These two incentives are:

- A desire to harm a designated target or group of targets.
- A desire to better one's (often the C&C master) monetary situation.

Concerning the first driving force, harming a designated target, a great many tools can be utilised to cause harm. One such method, as described in Kolias et al.'s paper [41], is through a Distributed Denial of Service attack (DDoS). This is showcased in the Mirai botnet back in 2016. Mirai, Japanese for "uture", was not the first botnet to emerge. As touched on in Osagie et al.'s paper [50], several botnets had already emerged, dating all the way from the late 1980s and early 1990s. It was, however, capable of performing an excessively powerful attack against the French webservice provider, OVH, with a peak throughput of 1.1 Tbps [51]. The reasoning for this attack, as it turned out, was based on the fact that OVH hosted a popular tool for Minecraft Server hosts [52]. Ironically, this tool helps to mitigate DDoS attacks against servers.

References [44–46] present some of the possibilities within the scope of monetary gain from botnets, either via actively cracking user credentials through various means or by cracking entire pieces of encrypted data. Another example would be barring the user from accessing a service or device they own or rely on, as documented by [53].

#### 4.1.1. Designated Targets

To understand the incentives behind the development of botnets, one must first understand the ubiquitous nature of botnets as a whole. Botnets may target a great many different objectives, sectors or groups in modern society, a natural conclusion given botnets' capacity to mobilise great numbers. The following unordered list of targets are but a handful of the potential victims and sought out results of botnets:

- Groups of political disparity or political critics, as discussed in Nazario's paper [42];
- National power grids and critical service providers, necessary infrastructure of modern day's increasingly technologically dependent societies, as described by Dabrowski et al. [54] and Sgouras et al. [43];
- Civilian peoples' information and passwords [44,45];
- Espionage and intelligence gathering of foreign nations [47];
- Cracking encrypted or hashed data [46].

The difference in targets of botnets is a great incentive in the development of botnets. They can target a broad range of victims, allowing the botnet master to either target whole groups of victims, or a single institute or individual. The versatile nature of botnets caters to a extensive list of use cases, leading to an ever growing demand for powerful, subtle and specialised breed of malware for botnet-based attacks.

#### 4.1.2. Reasons for Attack

As touched on briefly in the prior sections, botnets are developed and utilised for a number of use cases. Having gone over how diverse the targets of botnets may be, it is evident that the reasons must be just as diverse [10]. The same range of importance of targets is seen in the reasons for botnets, varying from the single user credentials for petty thieving to nation-spanning acts of terrorism.

Another major reason for the usage of IoT devices as the specific source of infection and attack of botnets is found in the very foundation of modern-day state of IoT. The devices are often mass produced using cheap, potentially outdated, components. While the capabilities of the devices are limited, they all have the ability to connect to the internet and perform some level of basic processing [31].

### *4.2. Benevolent Botnets*

While exceedingly rare, not all botnets are malicious. A scant few, such as the Hajime botnet, is an example of a neutral if not beneficial botnet [48]. Built on a similar method of infection as Mirai, Hajime distinguishes itself from its cousin in a number of ways, such as:

- A decentralised P2P distributed hash table, rather than Mirai's C&C approach.
- A far greater number of ways to infect new hosts.
- The usage of a custom made protocol for disseminating files.

Another interesting differene, is that is has never been used in a documented hostile attack on a service or platform. The only instances of potentially questionable actions performed by Hajime have been acts of broadening its sphere of influence to new IoT devices. In a remarkable act of selflessness, the botnet actively *patches* discovered security holes on infected devices, rendering many attack vectors used by other botnets mute. Other botnets are created by researchers to intentionally overtake and disable malicious botnets, propagating the harmless version instead [49].

## 5. Evolution of Botnets

For the purpose of clarification, Table 3 below details a number of papers discussed in this section.

Botnets, as a defined type of software, first saw the light of day in the late 1980s, with botnet toolkits going back to December 1993, with the release of the IRC-based Eggdrop [50]. Its original intention was for the C-based Eggdrop to be able to share data in between instances and act in a coordinated manner. While the original botnet was benevolent and served a honourable purpose, the derivatives have since been used for mostly malicious purposes, however. This section will go over papers and sources detailing the various differences and iterations a number of different botnets have gone through.

**Table 3.** Evolution of botnets and their associated papers. The first column describes the novelty and executive summary of the botnet evolution in question. The second and third columns explain the botnet attack vectors and the year of first mention. Lastly, the table lists the associated papers. Note: The papers listed is ordered by the year of the earliest documented occurrence of the described topic related to botnets.

| Associated Area of Interest | Vector of Attack | Year | Papers |
|---|---|---|---|
| First recorded appearance. | IRC forums. | Late 1980s | [50] |
| Honeypots is an often employed tool to detect botnets. New botnets have shown a capacity to identify and avoid detection from such measure. | N/A. | 2004 | [55,56] |
| An analysis and discussion of botnets based off of the Darknet. | Darknet. | 2006 | [57] |
| ZeuS botnet and its role as one of the most influential botnets in the world. | Various systems. | 2007 | [9] |
| Botnets have begun showcasing active methods and tools to circumvent detection. | Various means discussed. | 2007 | [7,58–60] |
| HTTP-based botnets are explored and discussed along with a multitude of different other botnets. | Browsers and extensions. | 2007 | [61] |
| Description of various botnet characteristics, the latest research and insight into botnets. | N/A. | 2009 | [62,63] |
| New type of botnet capable of impersonating human reaction patterns, a factor otherwise used to identify botnets typically. | Various systems. | 2009 | [64] |
| Smartphones have grown powerful enough to be a potential vector of attack, for a botnet. This is explored in detail. | Smartphones. | 2010 | [65–70] |
| Botnets as a service is a newly founded concept, and is explored in details. | Typical SaaS centers. | 2011 | [71] |
| New type of botnet structure, based around a P2P-oriented basis is investigated, discussed and analysed for potential vectors of attack. | None formally disclosed, architecture discussed instead. | 2011 | [72–77] |
| Other botnets use obfuscation tactics to hide the true identity/position of the C&C's location, showcasing a trend of botnets growing more versatile and elusive to researchers. | N/A. | 2013 | [78] |
| More kinds of botnet susceptible hosts become more common, leading to new potential vectors of attack. | Browsers, extensions, smartphones and online clipboards. | 2013 | [79,80] |
| Vehicles can also be a potential vector for botnets, such as GHOST. GHOST seeks out VANETs in cars to utilise the VANET control channel for communication. | Automobiles and other vehicles. | 2016 | [81] |
| IoT devices have become equipped with enough processing power to pose a sizeable threat. The generally poor safety implementations and the scale of IoT networks, makes them a good candidate for attack vectors. | IoT devices. | 2016 | [82–84] |
| Proposals for self-evolving botnets. | Unknown vulnerabilities in hosts. | 2016 | [85] |
| Cryptocurrencies have lead to explorations into new areas of potential botnets. Discussion and debate on the architecture. | Blockchain structures. | 2019 | [86] |

Botnets spanning hundreds of thousands of individual systems was a common sight in the early 2000's, with a few outliers in the millions of devices. The typical infection vector of insecure networking or lack of security updates have long passed, for new, more modern, more intricate and more obfuscated angles of attack [62].

In order to get a solid foundation on the state of modern day botnets and the threats they pose, Ogu et al.'s paper [63] from 2019 showcases some of the latest research and insight into the world of botnets. This consolidation of information is a great starting point for researchers looking into furthering their research on botnets and the issues the world faces in that regard. An interesting case of a recent wide spanning botnet is the ZeuS botnet. Etaher el al's paper [9] on ZeuS offers up an important explanation on one of the most influential botnets of today, with victims' losses in the region of hundreds of millions of dollars. ZeuS is an example of a botnet, which, with a staggering 3.6 million infected devices, proved extremely damaging to the American banking sector. As botnets become more commonplace, the availability of botnet-based attacks also increases for non-malicious actors. Botnets-as-a-service is a phenomenon that has also become common, allowing individuals to perform attacks such as DDoS without first developing and propagating their own botnet [71].

Finally, Sood et al. presents a recount of HTTP-based botnets in their paper [61], going over various botnets from ZeuS, SpyEye, ICE 1X, Citadel, Carberp, etc. The paper looks into the design and operation of these, summarising their findings in a list of various mitigation strategies.

### 5.1. Disguises and Subterfuge

In the early days, botnets would often attempt to avoid attention from authorities and government(s) by purposely avoiding targeting or utilising their systems. However, botnets have grown more and more clever and even capable of detecting a variety of detection mechanisms. Honeypots, devices purposely designed to be easy targets of botnets, can now be identified and avoided to help prevent detection [55,56].

Honeypot avoidance is not the only measure to avoid detection. Obfuscation of the C&C's location, as described by Wang et al.'s paper [78], highlights just one method of evasive action botnets may utilise. Botnets may also use dynamic IP ranges to quickly and easily circumvent IP blockages [58], or even fortify and defend its C&C center against Sybil and other routing table pollution attacks [7,59,60].

### 5.2. P2P-Based Botnets and Their Intricacies

As briefly mentioned previously in Section 4, some botnets utilise a P2P-based chain of command, over the usual C&C-based approach typical of botnets [72]. This decentralisation of the command structure helps to obfuscate the position of the commanding bot, as well as help defend against typical counter attacks against the botnet, such as key pollution from seized bots. Overall, this increases the resilience of P2P botnets manyfold, as no single-point-of-failure exists within the C&C structure [73]. This is explained in detail in Yan et al.'s paper [74], which also proposes a novel botnet called AntBot. AntBot is one of many new examples of more resilient botnets, showcasing the developments of this worrying trend. This type of hardened P2P-based botnet is also explored and explained in detail in Andriesse et al. [75]. In order to counteract this phenomenon in botnet evolution, entirely new approaches much be made, such as [76], which proposes a different take on detection of P2P botnets, based on its behaviour. Some papers, such as [77], have attempted to model the resilience of P2P botnets to help researchers identify weaknesses and potential mitigation against P2P botnets. These papers all try and tackle the developing threat.

### 5.3. Extension and Browser Based Botnets

Simple browser extensions for Google's Chrome or Mozilla's Firefox have in recent years seen a growing surge of interest from users. The ability to add additional functionality and capability to a browser, such sa blocking ads, easily downloading high-resolution images, etc. have made these small pieces of software an attractive tool. While the user's browser may be open about what permissions each individual extensions requires to function, the actual implementation and usage of these requirements are often uncharted territory to most users. This makes malicious browser extensions an excellent point of attack, as browsers often have permission to add, edit and delete files on the host system. This is showcased and documented in Perrotta and Hao's paper [79] from 2018. The paper's proposed extension-based botnet is but one take on a new variety of botnets, offering a number of different capabilities.

In a similar tone, massive online social media that connect people with one another have also grown vulnerable to modern botnets. This new breed of botnets, typically nicknamed Social Network Botnets (SNB), are capable of infiltrating deep into social networks such as Facebook without being caught or stopped by defence measures. Boshmaf et al.'s paper [80] details how such an SNB can be conceived and details how it performs on Facebook over a period of eight weeks.

Likewise, not only have social medias fallen prey to this new type of botnets. Online clipboards and publicly available cloud storage services have turned out to be effective measures to act as C&C centres for botnets, as described in Yin et al. [87]. Other examples include the proposed social botnet DR-SNBot by Yin et al., which argue that bots hiding within social networks are more resistant to to destruction compared to other types of botnets [60].

### 5.4. Smartphone-Based Botnets

As smartphones have grown more and more powerful and full of personal information, botnet creators increasingly look towards these pocket sized computers for new possibilities. Mobile botnets show disturbing results as a botnet vector of attack [65–67]. Interestingly, something as simple as an SMS sent from one smartphone to another can also prove to be highly potent, as some botnets have taken to this method to relay messages from the C&C to the bots [88,89].

Of further note within the field of mobile botnets, Malatras et al.'s taxonomy [68], and [69] by Rodriguez-Gomez et al. are both of great use to model and formalise botnets. There is also Pieterse and Olivier's paper [70] on this type of botnets, in which they present a valuable take on the evolution of this niche of botnets. All three papers provide excellent introduction and supplementary understanding of the various characteristics and interesting highlights of mobile botnets.

Smartphone services such as Googles Push Notification Service (PNS) is also considered to be exploited by botnet devs as C&C channels [90]. Android is not the only targeted OS, as seen in [91], where Apple's iPhone was the target of the iKee.B botnet, which collected system information such as SMS, network configuration, os name and os version.

### 5.5. Vehicular Botnets and Its Effect on Modern Traffic

Having touched on smartphone-based botnets, it is no surprise that vehicles are becoming increasingly vulnerable to botnet takeovers. Vehicular ad hoc networks (VANETs) are expected to play an increasing role in traffic safety as well as the driving experience. The ability for cars to communicate with one another may very well revolutionise the way people drive. VANETs are, however, under threat of new types of botnets, as touched on in [81].

### 5.6. Blockchain-Based Botnets

While blockchain has, for a large part, often been associated with cryptocurrencies, new methods and developments showcase a new type of botnets emerging based around blockchain. Bock et al. touches on this, in their assessment [86], providing a broad overview of the associated risks and relates the problems with this new type of botnets to existing C&C-based botnets.

### *5.7. IoT-Based Botnets*

Back in Section 4, a brief recount was made as to the reasons why IoT devices were especially popular botnet slaves. This is further explained, discussed and evaluated in a number of papers, including [82,83]. Situations such as poorly configured devices, the role of IoT in botnets as well as real life scenarios involving IoT devices capabilities for usage in attacks. Likewise, Mendes, Aloi and Pimenta's paper [84] on IoT based botnets offers great insight into various architectures employed by botnets.

### *5.8. Atypical New Botnet Variants*

Every once in a while, entirely new botnets pop up, bringing either new features, capabilities or counteractions to known botnet mitigation tools.

Chen et al. [64] discusses a new type of botnet, a so called 'Delay-Tolerant Botnet', pieces of botnet-enabling malware capable of impersonating human reaction times. This helps it avoid detection for longer, as reaction times are often a measure when identifying botnets and their attacks.

Abu Rajab et al. presents an analysis of a botnet within the Darknet [57], showcasing how botnets make up a substantial amount of internet traffic.

As a general model, Kudo et al. proposes the concept of self-evolving botnets [85] which models their behaviour through a stochastic epidemic model of botnet features. The behaviour of infection shows quick propagation and the model indicates that self-evolving botnets should be prevented from spreading early.

## 6. Detection and Mitigation

This section describes the botnet mitigation and detection strategies proposed within research. On the topic of detection and mitigation of botnets, the two components are often conjoined in research, as the mechanisms for detecting a botnet often correlates to its behaviour and infection vector. Through this, a mitigation strategy can be built to counteract the identified vector or behaviour, which either partially or completely nullifies the botnet. For that purpose, it was decided to follow the example from prior peers, and conjoin the two elements in this section as well. The distribution of papers and subcategories of detection papers can be seen in Figure 4.

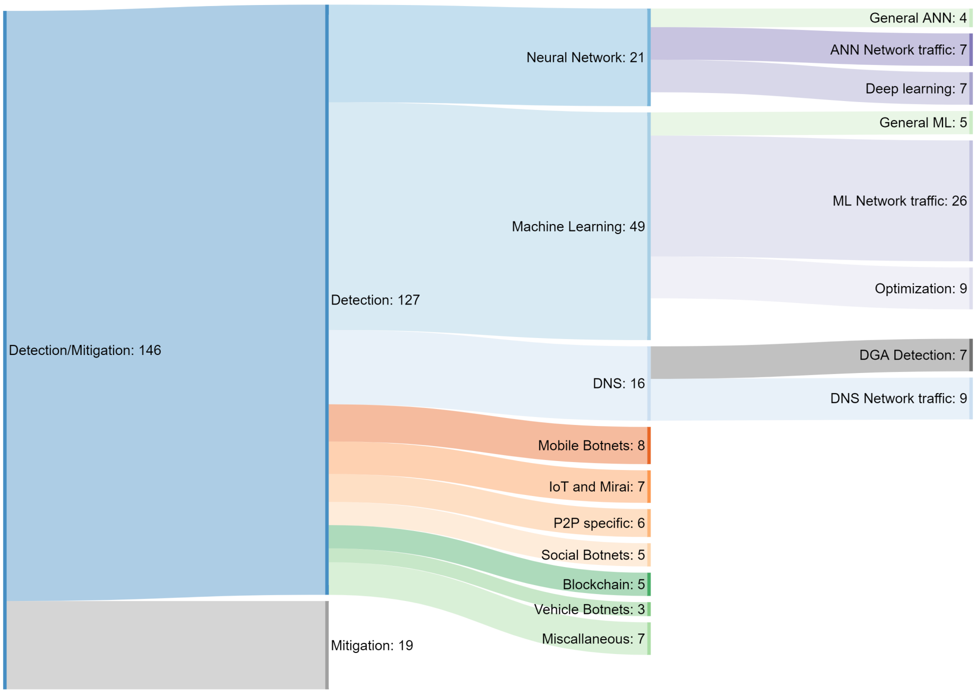

**Figure 4.** Distribution of botnet detection and mitigation mechanisms for this paper.

### *6.1. Detection Mechanisms—Techniques*

This section covers all papers, which are related to detection approaches and compares several techniques for each categories.

#### 6.1.1. Neural Network Detection Mechanisms

For the purpose of clarification, Table 4 above gives an overview of a number of neural network based detection techniques and their related papers:

**Table 4.** Papers describing detection of botnets using neural network based techniques. Each row describes the overall technique, known advantages, disadvantages, detection rate and related papers.

| Technique | Advantage(s) | Disadvantage(s) | Detection Rate | Papers |
|---|---|---|---|---|
| Back Propagation (BP). | Can detect botnets with no false positives as well as low expected error rate in higher error environments. | Only tested on a certain types of botnet traffic and characteristics of other botnets. | 99% detection rate, 95.7% accuracy and FP rates of 0.00952 or lower. | [92] |
| PSI-Graph | Much faster than FCGs, better FNR, FPR and accuracy. | N/A. | Accuracy of 98.7%, FNR 1.83% and FPR 0.78%. | [93] |
| Convolutional Neural Network (CNN) | Can automatically extract features of botnets and has higher accuracy than traditional NN. | Reference [94]: Training process requires GPU power. Reference [95] does not yet support transfer learning. | Best accuracy of ResNet is 99.32%. Reference [95] 99.98% accuracy for DenseNet and 83.15% for SVM. Reference [96] achieves up to 98.6% botnet detection accuracy on the self-tests and about 90% on the cross-evaluation test. | [94–97] |
| Artificial Neural Network (ANN) & MLP-ANN. | Reference [98] has low computational overhead. Reference [99] uses supervised learning approach to obtain high TPR and can further be used in Transfer Learning projects. | Reference [99] does not have hybrid models. | Reference [98] managed to get minimum of 87.56% TPR during testing. Reference [99] got 100% accuracy and TPR. | [98,99] |
| ML & DL. | A combination of multiple ML and DL models including comparisons. | N/A. | N/A. | [97] |
| NN & AIS. | Reference [29] can provide endpoint protection. Reference [100] does not need prior knowledge of botnets. | Reference [29] requires command and control server. | N/A. | [29,100] |
| Machine Learning, Deep Learning, t-distributed stochastic neighbor embedding & Deep Neural Network. | Considers a broad amount of ML and DL techniques. | N/A. | DNN takes a long time to train and also around 1.3 s to execute detection logic where other methods found in this paper is faster or requires less training time. | [101–103] |
| Nonnegative Tucker decomposition. | Memory-efficient. | Normally requires too high a computational cost to run in real time. | N/A. | [104] |
| NN with blockchains. | Lightweight: small memory and low-power processors needed for devices. | N/A. | N/A. | [29] |

Neural network-based detection of botnets is just one of many proposed methods of botnet detection. X.G. Li and J.F. Wang [92] proposes using back propagation (BP) neural network to detect botnets based on traffic characteristics. Other detection methods, such as the one proposed by [93], also use similar neural network methods for detecting IoT-based botnets using PSI-Graph generation with potentially fewer resources. Reference [99] uses a model based neural network approach to classify IoT botnets; the paper compares the MLP-ANN mode with the N-BaIoT model. MLP-ANN requires a supervised learning approach, meaning it can become even more effective by training with more data and can run on very limited computing resources. N-BaIoT on the other hand works unsupervised (USML) but requires a larger resource overhead. Reference [100] uses a biology-inspired artificial immune system approach to model botnets as infections within a network body. The microorganisms within the artificial immune system are trained to act upon spam and scanning related botnet activity.

Other papers focus more on applying neural networks to detect irregularities within network traffic. Dhalka et al. compares several contemporary botnet detection techniques, k-means clustering, neural network and recurrent neural network. Their paper [105] compares the algorithms in terms of several factors, including positive/negative rates, sensitivity, specificity and more. The paper identifies the neural network method as the best solution based on the chosen measures, with a caveat that the neural network method may not be practical. There has been a growth in papers related to mitigating botnets found in IoT devices, as this industry is growing exponentially without regards to security. Alexander and Allison Nixon propose an Industry Security Association committee to be created and publish security standards which manufactures are required to follow [106].

For non-IoT botnets, other papers such as [25] address four different major botnet detection approaches: signature-based, anomaly-based, DNS-based and mining based detection. The paper evaluates previous surveys and illustrates botnets architectures, topologies, communication protocols, attacking method and, their destinations, impediment approaches, and detection techniques. Similar neural network identification systems such as [107] work by analysing botnet traffic, using a more adaptive and flexible stream mining algorithm to classify botnets. Reference [94] also proposes a similar network analysis approach with a neural network-based P2P model to monitor botnet traffic and recognise patterns using the ResNet architecture. In another similar approach, the neural network-based detection and mitigation system called BoNeSSy also analyses network traffic to detect and mitigate botnet behaviour [98]. If an application identifies a threat, BoNeSSy will notify the administrator and take appropriate security actions to isolate the potential threat. Chu et al. proposes a combination of machine learning and classification mining [108] for botnet detection.

Jithu et al. [102] propose a deep learning method that detects botnets in IoT devices using anomaly detection. The technique employed in the paper reaches an accuracy of 94% and recognises the need for IoT security with a predicted number of 24.1 billion IoT devices by 2030. Abdullah et al. [103] propose using a Local Global Best Bat Algorithm with neural networks (LGBA-NN), which achieves a 99.89% accuracy in their study using the N-BaIoT dataset. Their study includes comparing LGBA-NN with less effective implementations of PSO-NN and BA-NN.

Deep learning, which employs neural networks, has been used by Taheri et al. [95], who proposes a deep learning-based botnet detection engine that takes raw network traffic data as input and transforms them into images. These images are then input into a deep convolutional neural network (CNN), DenseNet, for classification of normal and botnet traffic data. CNN approaches are endorsed by [24] as being one of the best performing techniques for detecting botnets in IoT devices along with Recurrent Neural Network (RNN) and Artificial Neural Network (ANN). A similar approach to [24] is using deep learning to construct algorithms to detect IoT-based botnets and botnet attacks. Sriram et al. [101] propose an algorithm that analyses the network flow and can be used to secure “smart city applications”. This includes health care, power grid infrastructure, water treatment facilities, traffic controlling, etc. Additionally, the flow of networks can be utilised for further analysis and learning, to enhance the performance of the algorithm. The authors of "Real-time botnet detection using non-negative tucker decomposition" [104], propose a method for detecting group activities from extracted features in darknet traffic using tensor factorisation. While this method requires too high computational costs to run in real time, they propose implementing a two-step algorithm in order to achieve fast, memory-efficient factorisation. More nontraditional methods like [96] seek to identify botnets through the usage of power consumption as the parameter for their CNN model. In [29] a similar lightweight solution is also mentioned for use in small memory capacity devices with low-power processors, since these are not able to have reliable anti-malware systems. It is based on the use of NeuroMesh, which is a combination of neural and mesh detection networks used to secure the devices. It can detect and delete malware and implements IP-based blacklist and whitelist access control to provide secure channel for IoT devices via the Bitcoin communication protocol.

#### 6.1.2. Machine Learning and Network-Based Detection Mechanisms

For the purpose of clarification, Table 5 below details a number of papers that goes over machine learning-based detection:

**Table 5.** Papers describing detection of botnets using machine learning based techniques. Each column describes the overall technique, known advantages, disadvantages, detection rate and related papers respectively.

| Technique | Advantage(s) | Disadvantage(s) | Detection Rate | Papers |
|---|---|---|---|---|
| Network flow analysis | Real-time detection. High detection rate (up to 98%) of known signatures. Can be implemented in current SDN solutions. Supports a wide variety of detection approaches. | Requires training data for unknown attacks. Too many features in selection results in unnecessary overhead. | Varies between 85.34%. Reference [109] to >99% [26]. | [26,28,97, 109–129] |
| Honeypots | Ability to train model on unknown variants. | Training needs to be supplemented by simulated network traffic. | 99%. | [130,131] |
| DNS-based profiling of Mirai botnets | Uses live datasets based on honeypot infected botnets, low computational time (<0.2 s). | Limited to mirai(-like) botnet variants. | >99%. | [132] |
| Cloud-based detection offload | Linear scaling with number of computational hosts. | Real-time detection not possible. | N/A. | [133] |
| Minimization of ML feature selection | Lower computational cost while keeping high accuracy rate. | Limited to IoT-based botnets ([134]), does not improve weaknesses to unknown signatures. | 98.97% ([134]) and 75–99% ([135]). | [134,135] |
| VM Hypervisor detection agent | Allows OS-level passive detection and monitoring. | Only applicable for VM-based hosts. | N/A. | [5] |
| Self-adaptive system with fuzzy c-means clustering | Can choose security scenarios and adapt mitigation procedures depending on the attack. High resilience. | Dependent on the specific system network used during testing. | High percentage of known and unknown. Multi-vector cyberattacks: 70%. | [136] |
| Network botnet fingerprinting and signature | Large data throughput. High accuracy. Reference [137]: better real-time performance compared to CPU-based approaches (800–1300% speedup) | Specific parameters in the used dataset, which need more features. Reference [137]: requires dedicated GPU hardware. | Very few false positives. Accuracy close to 100%. | [137,138] |

**Table 5.** *Cont.*

| Technique | Advantage(s) | Disadvantage(s) | Detection Rate | Papers |
|---|---|---|---|---|
| Network traffic data mining | Allows for detection of botnets inside complex traffic before the attack. Does not require network changes. | Method needs to be deployed by ISPs. Difficulties when using the NAT technology. | Detection: 98%<br>NaiveBayes: 89%<br>BayesNet: 87%. | [139] |
| Multi-phase traffic ranking mechanism | Reduces the false classification rate of normal IP traffic. | Needs further work to detect different kind of HTTP botnets. The experiment data is limited. | N/A. | [140] |
| Multi clusters for classification | Achieve high accuracy and reliability. Outperforms individual clustering algorithms in training time. | Increased algorithm runtime complexity. | N/A. | [141,142] |
| ML on DNS query data | Better than IDS based detection on newer botnets variants. | Takes time to train. | Most ML algorithms score over 85% accuracy on DGA botnets, among which the random forest algorithm gives the best results with an overall classification accuracy of 90.80%. | [143] |

Neural network is not the only method to use the N-BaIoT dataset, as seen in [97], where Bashlite and Mirai found their way into various IoT devices. These included doorbells, baby monitors, security cameras and a webcam. Detection models were developed for each device using numerous machine learning modes, including deep learning models. Similar machine learning methods have been used by Long Mai and Dong Kun Noh [141] using cluster ensembles to increase detection reliability compared to other clustering mechanisms. Instead of classifying flow clusters in either a botnet flow or normal flow, the algorithm uses multiple clusters for the same traffic and a link algorithm to do the final classification. Self-adapting systems for detecting, clustering and classification of botnets is proposed by Lysenko et al. [136], who use a semi-supervised fuzzy c-means clustering technique. The system is also able to double as mitigation as it can reconfigure corporate networks and execute more specific actions such as reducing request timeouts, decreasing allowed HTTP request size and blocking source hostname and IP addresses. Reference [142] also applies a clustering machine learning algorithm to detect Internet Relay Chat (IRC) traffic containing botnet behaviour. The approach however is based off a fuzzy cross association clustering algorithm to study the relationship between known traffic and unknown traffic. Unknown traffic can then be checked to verify or disprove the appearance of a botnet within the IRC traffic. Machine learning can be very helpful when it comes to detecting different kinds of botnets, but recently, bot herders [144] have begun to use well-crafted concept drifts based on known machine learning techniques to defend against ML assisted detection.

Through a new ML algorithm consisting of a combination of ANN and DT, Rezaei [145], has obtained a detection accuracy of 100%. The technique has a noticeable 11.36 s duration detection time using 20 features to detect botnets in IoT. Seungjin et al. [146] refers to what they call smart factory (SF), which is a combination of AI and ML. They tested two different ML techniques, Weka and R-studio, achieving 95.3% and 96% accuracy, respectively. Pandey et al. [126] use RF to classify the data into multiple units and then SVM to reclassify every sub-entity to improve accuracy. Their RF-SVM hybrid ML model achieved 85.3% accuracy while RF-Naive Bayers reached 83.36% and lastly RF-KNN-LR 79.56% accuracy.

Hidayah et al. [147] obtained up to 92% accuracy using ML algorithms that filter and classify data to detect the botnets C&C server. Siqlang et al. [148] studied the use of unsupervised detection of botnet activities and used the Frequent pattern tree algorithm provided by Weka. They achieved up to 100% accuracy varying with the thresholds chosen and up to 100% precision. Mehdi [149] found that using both ML and DL techniques based on a somewhat hybrid combination of cooperative game theory, accuracy and learning times could greatly be improved. For SVM, he obtained 11.62% improved accuracy and 154.41 s better learning time and for LSTM, 0.24% better accuracy and 222.72 s better learning time. Mehdi also found that these methods achieved an accuracy of 99.98% and higher using 10 or more features for detection.

Using KNN, Bjatt et al. [150] achieved in scenarios up to 98% accuracy and provided comparisons to other methods such as Spark-ELM, CCD and Bclus. The proposed method detects botnets based on a forecastive anomaly detection approach, where the first progression is the instance creation and the second is Cataloging. After the progressions, they use Graph Structure Based Detection of Anomaly (GSBDA) to detect hazardous anomalies and lastly use a KNN to identify the botnet accurately. Ali and Fatemeh [151] uses DNS queries to extract features from network traffic and then apply ML to generate a botnet detection report. Their studies included testing DT, SVM, RF and Logical regression as their ML algorithms and obtained accuracies of 98%, 96%, 99% and 93% respectively. Panda et al. [152] claim 100% accuracy using two different approaches, the first approach is scatter search (ScS) combined with CNN and the other method is ScS combined with Deep Multilayer perceptron (DMLP). They tested their implementation on the UNSW-NB15 dataset. where 66% of the data were used for training and the remaining 34% for testing.

Another general category within machine learning algorithms is the use of network anomaly [26] focused algorithms. This kind of mechanism of clustering with machine learning can be found in [138], where a new method called BotFingerPrint (BotFP) is presented. BotFP is supposed to be a more lightweight method that can handle a large number of data easily. BotFP is also designed to detect malicious network activities such as port scans and DDoS attacks. Kozik and Choraś introduce techniques [124] used in big data and machine learning to identify botnet traffic in networks. The multi-scale analysis model is used to extract botnet features from network traffic, which are then classified using a random forest machine learning algorithm. Poisson sampling is further used to train the random forest model by under-sampling benign traffic. Chen et al. [125] propose a method similar to Kozik and Choraś [124] with a conversation-based detection mechanism by using a random forest algorithm to classify botnet conversations in network flows. Conversations are classified depending on their duration, size and distribution of topics. The random forest algorithm is used for selection of probable botnet flows for detection using a separate machine learning algorithm trained with random forest. Besides using random forest, Reference [126] found Support Vector Machine (SVM), Naive Bayes (NB), K-Nearest Neighbour and Linear Regression algorithms to be possible detection mechanisms. Furthermore, Reference [109] conducted an analysis of various machine learning algorithms for botnet DDoS attack detection, including SVM, ANN, NB, Decision Tree (DT) and USML. According to [109], when considering only DDoS attacks, Unsupervised Learning (USML) stands out as the better option to differentiate between botnet traffic and legitimate network traffic.

Kirubavathi and Anitha also present an approach for detecting botnets through network traffic flow behaviour analysis and machine learning. The proposed method [127] extracts network features such as small packets, packet ratio, initial packet length and bot-response packets. The data are then classified using three machine learning algorithms, Boosted Decision Tree (DT), Naive Bayesian (NB) Classifier and Support Vector Machine (SVM) to classify benign and botnet traffic. In common with Kirubavathi and Anitha, Lin et al. [139] propose a method to identify P2P botnet traffic using data mining on network traffic with NB algorithm. Furthermore, Reference [23] proposes detection, prevention, investigation and mitigation using anomaly, signature, DNS, data mining and hybrid techniques. Lin et al. also proposes the use of J48 and Bayesian networks to be applied to the monitored traffic data, while Lee et al. addresses the use of a ranking algorithm to clustering-based botnet detection algorithms [140]. The ranking algorithm gives a higher ranking for source/destination IP pairs with identified suspicious behaviour. The paper argues that only using k-means clustering results in a large degree of false positives, and that the problem can be solved by ranking the resulting clusters by suspicious TCP and ICMP traffic per source/destination IP pair. Further endorsing the use of k-means, Li et al., propose a botnet detection mechanism using the particle swarm optimisation and K-means algorithms to identify botnet network behaviour [128]. Su et al. proposes a machine learning approach to detect P2P botnets in software-defined networks (SDN) [129]. Detection results are provided to an OpenFlow controller in the SDN, which creates rules to control how botnet source packets are handled at the network switching level.

Along with network analysis, filters can be applied to help extract relevant features from network traffic such as connection duration, service type, connection state and more. In [110] by Indre and Lemnaru, the features are provided to a static filter, binary classification filter and a malware detection filter. These filters can reject the connection based on static header rules, general behaviour logic and specific cyber-attack detection, respectively. Also acting on network behaviour and feature set extraction are multiple papers [111–114], which propose detecting HTTP-based Command & Control servers using behavioural analysis. The feature set found by the papers can be used to further train machine learning algorithms to become even better. Other papers make use of similar methods. Reference [28] uses a supervised machine learning algorithm using a random forest classifier to identify anomalies in IoT networks. Reference [115] proposes the use of SoftFlow to capture packages and generate NetFlow for machine learning. The paper applied this method to two botnet datasets to test if the method was able to differentiate between legitimate Alexa traffic, Citadel and Zeus botnet traffic. More methods based on existing industry frameworks have also been tested. References [116,117] use Cisco's Netflow for analysis along with a custom-made detection framework to detect botnets. The botnet propagation model uses a modified Susceptible, Infectious or Recovered SIRS epidemiological model to estimate if there will be an epidemic of the given botnet and then uses the developed framework to mitigate the infection.

Moving into machine learning combined with the use of honeypots to detect botnet-enabling malware. Ruchi and Kumar [130] proposes using ThingPot which is a virtual IoT honeypot capable of catching various botnet binaries by emulating different IoT communication protocols along with entire IoT platform behaviours. However, with honeypots becoming more normal in the line of defence against botnets, bot herders also become better at bypassing them. Therefore, Reference [131] seeks to make honeypots more efficient and more effective. Owen et al. seeks to use DNS traffic analysis models with a profiling scheme of Mirai-like botnet activity captured globally in distributed honeypots [132]. It discusses features useful in profiling botnets in the past and suggests a number of improvements. The suggested solution can bring down botnet detection time significantly while maintaining high levels of accuracy under random forest formulation.

A great amount of botnet detection mechanisms, most of which are based on network analysis, will not use real time detection, as the high number of data overwhelm most CPU detection-based systems. Because of this, Che-Lun and Hsiao-Hsi propose the use of GPU based detection over CPU-based detection to gain a speedup in real time detection [137]. By using GPU based detection, packet loss would occur less frequently as the throughput capacity of the detection system increases. This allows for a very noticeable speedup. Using an approach designed to reach near real-time detection, but without the speedup benefit proposed by Che-Lun and Hsiao-Hsi, Reference [118] seeks to detect Command & Control servers using autonomous methods. This method eliminates the need to manually detect C&C signatures from an intrusion detection system (IDS). GNU Anubis, which is an SMTP message submission daemon, feeds all the IDS data and extracts all frequent strings. Then, a ranking function will assign high scores to traffic-class-distinguished strings, as these are more likely to be good C&C signatures. The authors conclude that the method is a meaningful way to extract C&C signatures in real-world applications.

In other near real-time detection mechanisms, Reference [119] proposes an open-source network-based botnet detection and mitigation tool called BotFlex. The tool functions as an intrusion detection system (IDS), passively listening to network traffic and determining botnet traffic from various parameters such as blacklists, C&C detection, outbound spam and more. Toby J. Richer [120] introduces an entropy-based detection mechanism to better detect botnet traffic with variance in beacons to C&C servers. The introduction of an entropy-based measure of delay variance allows for the detection of both fixed-delay and variable-delay beacons. As Sadhan and Moura experimented with tinyP2P and SLINGbot to detect periodic botnet behaviour in botnet traffic by analysing control plane traffic [121]. A somewhat similar approach is BotGM [122], which identifies network traffic behaviour using graph-based mining techniques to detect botnet behaviour. The approach also models the dependencies among network flows to trace back to the root botnet propagators. A study done by Rui et al. [123] shows the behaviour of the Grum, Cutwail and Bobax botnet. The study shows that once a host is infected, a number of Unknown TCP packets are sent on port 80 (in fact HTTP traffic). After multiple SIP invite packets and NBNS queries, the bots usually change a bit in behaviour. The bots behaved like expected with unknown UDP traffic as well as a high amount of HTTP traffic, DNS traffic and SMTP packets for DoS attacks. Their study also shows that these bots mostly infect countries in Europe and America. Supporting this is [5], which further proves the effectiveness of behaviour-based detection systems. On a virtual machine (VM), a detection agent is installed, which monitors the processes and their spawned processes to build a behaviour profile and bot process activity log(s). Calculating the Jaccard similarity coefficient between the behaviour profile and process activity logs is used to indicate if the host is infected or not. In their experiment, they show that their bot behaviour profiles and passive detection agent can distinguish bot hosts with no false positives and no false positives.

Other research, Reference [133], has also focused on increasing the throughput of real-time DNS-based botnet detection mechanisms. The paper in question proposes offloading fuzzy pattern recognition of suspected botnet traffic to the cloud, executing the detection in parallel and allowing for near real-time detection. Hoang and Ngyuen have tested several machine learning approaches for domain name systems (DNS) botnet detection, finding random forest to be the best choice [143] when it comes to use DNS query data. References [134,135] further propose the reduction in the network features used for detecting botnet traffic in order to speed up the detection process. A feature minimisation exercise shows the possibility to reduce the selected feature set while still providing a high degree of precision.

### 6.1.3. Domain Name System (DNS) Based Detection

For the purpose of clarification, Table 6 below details a number of papers that cover DNS-based detection.

**Table 6.** Papers describing detection of botnets using DNS-based techniques. Each column describes the overall technique, known advantages, disadvantages, detection rate and related papers, respectively.

| Technique | Advantage(s) | Disadvantage(s) | Detection Rate | Papers |
|---|---|---|---|---|
| DNS traffic monitoring | Can detect known and unknown botnets by monitoring DNS traffic anomalies. Can detect C&C server migration. | The huge size of traffic occurring in network environments is computationally intensive, and due to this, most DNS traffic monitoring methods are also not real-time. | Positive rate  90% and negative positive rate  5%. Reference [153] managed accuracy of 95% with 0.1% false positives. Reference [154] archived 98.52% True positive as minimum and 0.39% False positive as highest. | [22,23,25,153–158] |

**Table 6.** *Cont.*

| Technique | Advantage(s) | Disadvantage(s) | Detection Rate | Papers |
|---|---|---|---|---|
| DNS traffic monitoring assisted by mining | Can detect known and unknown bots. Can successfully locate C&C traffic. Most methods have low false positive rates and can detect encrypted communication. | Not real-time. | Random forest can archive up to 99% accuracy. | [22,23,25,132] |
| Network anomalies | Can detect known and unknown bots. | Not useful for detecting C&C traffic. Not real-time viable. | Positive rate of close to 95% and False positive rate lower than 3.5%. | [22,23,25,133,159] |
| Signature-based detection | Can detect most known simple botnets. | Cannot detect unknown or more advanced known botnets. | N/A. | [22,23,25,159] |
| Feature-based detection (CAFE/CSTA) | Can reduce the number of non-active C&C suspected Domain Names by 79.96% with false positive rate of 0.69%. | Can be affected by malformed DNS answers or DNS cache poisoning attacks. | N/A. | [160] |
| Support vector machine SVM | Can detect existence of botnets in small to medium sized networks. | Anti-malware software can increase the false positive rate. | Detection rate of 0.935 and false positive rate of 0.02. | [161] |

Domain Name System (DNS)-based detection algorithms are another frequently used approach to combat botnet threats. Most DNS approaches use an allow– deny-list concept to distinguish Domain generation algorithm (DGA) botnets from legitimate traffic. This method is used in [158], where it is seen that most of the domains and their traffic will be allowed by the list. Meanwhile, the rest of the traffic will be clustered using the density-based spatial clustering of applications with the DBSCAN algorithm. The clusters are further analysed to identify botnet domains. This method is similar to [153], which tries to detect botnet-based DNS traffic by the use of Power Spectral Density analysis, a signal processing technique. The method used in [158] shares many similarities with [162], which further adds the use of botnet-generated domain names identification using entropy measurements and n-gram scores. The domain names are then measured using a k-means clustering algorithm to identify domains which are likely to be generated by the same botnet DGA. References [163,164] instead use the lexical properties and semantic patterns of real domain names to train their proposed detection schemes. Wang et al. [154] exploit the behaviour of DGA botnets to identify potential botnet traffic. Botnets have a high number of failed DNS lookups stemming from the use of DGA algorithms to generate domain names. The algorithm filters botnet-generated domain names and clusters them using the Chinese Whispers algorithm. The clusters are then classified using a supervised machine learning algorithm, based on DNS query times and query amount. Truong and Cheng take several of these algorithms and compares their ability to detect DGA based botnets. Their paper [165] includes a comparison of Naive Bayes, K-nearest neighbour, random forest, support vector machine and decision tree. A more specific usage of Naive Bayes along with AdaBoost, C4.5 and SVM for Flickr profiling as proposed by Natarajan et al. [166]. The multilevel social network profile analysis method is used to detect the Stegobot on social networking websites along with identifying a range of image malware, botcargo and stego images used to identify Stegobot. Reference [167] proposes a new method of combined detection, mitigation and clean-up for next-generation botnet combating. The system consists of five modules with a task each. This system should be able to communicate, report, detect and heal itself when botnet-enabling malware enters the system. Detection is based on DNS host files and network inbound ports, which are analysed by the administrator along with a MD5 checksum of the tcp.sys file.

Monitoring activity from DNS-queries during C&C communication or updates and applying semi-supervised fuzzy c-means clustering to produce security scenarios is the basis of the self-adaptive system called BotGRABBER [161]. Not much different is the method proposed by Sharalfaldin et al. in [168], where a novel botnet detection framework, BotViz, is presented. BotViz uses a combination of DNS-based analysis of host PC DNS records and API hook forensics on memory dumps to detect potentially vulnerable systems. Forensics are done through an analysis module that uses a k-cluster machine learning algorithm to decide whether or not a host might be compromised by a botnet. Other papers seek to develop methods for botnet detection based on botnet behaviour called C&C Tracer. The C&C Tracer [160] works by using C&C active behaviour feature extracting (CAFE), domain name status querying (DNSQ) and C&C status tracing analyser (CSTA) along with allow lists from multiple external sources such as the Honeypot project and Shadowserver Foundation. An analysis done by Ichise et al. [156] to test the feasibility of botnet detection through domain name system (DNS) records. The analysis shows that in the 5.5 million DNS TXT record queries obtained from their campus network, around 2293 queries where classified as "unconfirmed". In their further investigation, ~22% of these queries were targeting suspicious URLs identified by virustotal [169]. A similar approach is used by Jin et al. in [157], which proposes a novel DNS-based detection approach for detecting botnet activity. The paper focuses on direct outbound DNS queries on non-standard authoritative name servers to identify botnets, which use TXT records to send commands. The paper finds that a similar 19% of identified potentially malicious DNS queries have been flagged by online websites, such as [169], for being used for botnet activity. Reference [159] also proposes a similar idea, but with a focus on UDP network traffic, focusing on DNS MX queries, the DNS packet request and various behaviour that might be botnet attacks based on UDP traffic. Reference [170] talks about a profiling dataset. "UMUDGA: a dataset for profiling DGA-based botnet" aims to enable researchers to move the data collection, organisation and pre-processing phases forward. Ensuring the availability of good datasets also help the general research community in providing novel detection mechanisms.

#### 6.1.4. Detection Mechanisms—Pervasive Computing Paradigms

The segment highlights different detection techniques employed in various types of pervasive computing paradigms. These paradigms show different ways in which hosts can establish communication channels and networks, which also affect how botnets can be detected within those networks.

#### 6.1.5. IoT and P2P Botnets

With the Mirai attack in 2016, some focus have shifted towards IoT networks as potential vulnerable hosts for botnet infection. Therefore, multiple mechanisms for IoT botnet detection have been proposed (see Table 7), both specifically against Mirai and also some more general mechanisms [171]. Reference [172] specifically targets Mirai and other known types of attacks with a quantum-inspired detection algorithm. The algorithm matches network traffic headers with a predefined table of IoT botnet attack signatures to detect malicious packets. The authors acknowledge that while not all kinds of botnet attacks have been considered in their approach, the method shows very high true positive rate for detecting known types of IoT botnet behaviour. Reference [173] proposes a sparse-representation framework for botnet detection on the IoT edge. The sparse-representation factor is determined from the network traffic of each individual IoT device, which is then compared against a threshold to determine potential malicious traffic. This allows the network controller to cut off any potentially infected IoT devices. Reference [174] argues for the use of logistic regression of IoT traffic to calculate the probability of an infected device. The regression is based on multiple network parameters including ports, number of requests, mean packet size and more. Finally, Reference [175] proposes using a local agent on IoT devices in an installation to collaboratively compute security events to detect botnet attacks. Botnet attacks are determined on the basis of the difference in DDoS traffic and benign network traffic, which is collectively decided upon by the agents.

**Table 7.** Papers describing detection of botnets in IoT and P2P-based environments. Each column describes the overall technique, known advantages, disadvantages, detection rate and related papers, respectively.

| Technique | Advantage(s) | Disadvantage(s) | Detection Rate | Papers |
|---|---|---|---|---|
| Quantum computing to combat Mirai | Fast results. High accuracy. | Only targets botnet with known signatures. | N/A. | [172] |
| Sparse-representation framework on IoT | Faster than compared approaches. | Tested on limited IoT botnet dataset. | 90%. | [173] |
| Logistic regression of traffic | High accuracy. High precision. | Can only detect during propagation phase. | >99%. | [174] |
| Collaborative multi-agent | Can detect large scale DDoS attacks. Lightweight: can be installed on hardware with limited resources. | Needs a minimum level of collaboration across organisations. Training environment does not reflect real-world environments: A percentage of agents may not acting as excepted. | Depends on framework implementation. | [175–177] |
| Bitcoin Miners detection | Experiments are showing excellent accuracy. | Only applicable on specific botnet types. | Depending on the variety of techniques that are utilised. | [178] |
| Classification from traffic frequency and behaviour. | No statistical traffic patterns need to be known in advance. Does not rely on payload data. Does not require monitoring of individual host. Effective (very high detection rate). Scalable low false positive rate. | Requires installation at all network boundaries. Unable to detect botnets with low amount of requests. | Up to 100%. | [76,179–187] |
| Graph-based approach | Computationally efficient. | Needs a certain amount of training data. Accuracy depends on datasets inspired of the evolving Internet state. | N/A. | [188] |
| PageRank algorithm using clustered cloud computing | Efficient performance (clustering). Scalable and Effective. Easily usable at low costs. | High computational needs. | 99%. | [189] |
| SMTP analysis | Can catch both text and image-based botnet spam mails. | Limited to mail-based botnets. | Detects 96.23% of botnets spam mails with no false positive. | [190,191] |
| Botnet application sandboxing | Computationally cheaper compared to contemporary intrusion detection systems. | Legitimate emails can be flagged wrongly as spam. | Not tested | [192] |
| Evidential reasoning | Can improve botnet detection rates. | Lacks an uncertainty evaluation model. | Up to 90%. | [193] |

Blockchains are another useful paradigm, which can be included in botnet detection techniques. In the paper [176], the use of lightweight agents included in many IoT installations is discussed. The main goal is to provide a secure communication channel, such as a private network, between each node (agent). Since all agents are able to communicate with each other, they can exchange relevant information such as collection of traffic metrics to identify ongoing DDoS attacks and victims. This exchange procedure is implemented via a blockchain smart contract, which is co-maintained by all nodes in the system. The involved blockchain technology ensures integrity among all the nodes and allows for the collaboration of the distributed nodes without a need of a third party. Another purpose of this technology is in [177], where blockchain is implemented as a framework using HyperLedger to give traceability of the hardware. By the use of a physically unclonable function (PUF), all the IoT-connected devices are sure to be unique. In this way, blockchains is used for verification in order to compare and identify these devices with their unique fingerprint ID. Among all these papers, the blockchain-based structure is often used for integrity, decentralisation and transparency among the participants of the chain. It allows these agents to communicate in a much more secure way and therefore make the detection mechanism more reliable and efficient. However, according to a different perspective, A. Zareh and H. R. Shahriari detail another type of target in [178], namely called "botcoins", which are Bitcoin miner botnets. They propose the use of dynamic analysis of instruction traces in suspicious executable binary files. A constant parameter value in the assembly exist in all botcoin implementations, which can be detected at the assembly language level. Compared to the other blockchain approaches, the detection strategy in [178] does not iuse blockchain as a communication channel, but analyses how a specific type of botnet functions.

The approach described in [180] uses two factors to determine if a P2P host is part of a botnet: host living-time and command search frequency. The paper argues that P2P botnets exhibit longer-term peer connections and high search request frequency compared to benign P2P traffic. Because legitimate P2P peer connection time is usually short and pull-style communication is uncommon, botnet-behaviour can be detected by those two factors. Likewise, Reference [181] also approaches the detection of P2P botnets by the P2P search frequency. The detection mechanism specified in the paper also considers the number of P2P peers, the argument being that P2P botnets have a larger number of peer connections compared to normal P2P traffic. The paper also looks at the periodicity of messages sent, with the argument being that bots periodically request commands from the botmaster. In [189], detecting peer-to-peer botnets using a high-level abstraction of parallel computing called MapReduce is discussed. MapReduce is aiming to divide input data into multiple inputs to make applying functions easier to them. MapReduce also helps running the tasks in parallel over multiple servers. Reference [182] uses both periodicity and active peer connections to determine if a host is part of a P2P botnet. The mechanism described also looks at the ratio of small packets vs. large packets to indicate C&C queries by bot hosts. Other papers such as [183] propose the use of of traditional network traffic analysis based on packet feature selection to detect P2P-based botnets. *PeerHunter* looks at the number of mutual peers between hosts to detect P2P botnet participants. The number of mutually connected nodes indicates the number of potential botnet communities and is used to identify candidates for botnet detection [179]. Reference [76] further builds on top of *PeerHunter* to identify whether the previously identified communities are part of a botnet or not. The detection mechanism for the communities use a network flow analysis method to detect botnets, with the primary factors being the ratio of egress/ingress packets, mutual contacts ratio and destination diversity ratio. Reference [188] uses a graph-based approach to detect P2P traffic, instead opting to exploit the structural properties of the botnet P2P overlay network. The approach checks the number and size of weakly-connected components, average node degree and InO ratio of the P2P overlay network graph to determine if the P2P network is a botnet.

Reference [185] uses firewall logs and the number of outbound connections to detect botnet behaviour. If the number of outbound connections suddenly increases above a threshold, the user is informed. Reference [184] detects HTTP botnet traffic in streaming logs by the use of Lanczos method. The log entries and time slots are put into a matrix to check for correlation with botnet behavioural traffic. The paper primarily focuses on comparison to principal component analysis (PCA) and shows that Lanczos method achieves similar results with a 25% reduction in runtime compared to similar approaches. Reference [186] proposes a multi-faceted detection mechanism based on both host and network analysis. The network analysis is based on known botnet behaviour while host analysis is based on the expected host processes and behaviour. If behaviour exceeds or goes beyond expected thresholds, the approach assumes that botnet activity is happening. Reference [187] proposes a method for detecting HTTP based botnets and C&C communication in the cloud using traffic analysis. The paper looks at five instances of packet capture and analyses the HTTP (TCP packets) traffic, calculating entropy of the captures with TCP payload, length of payload and frequency of each character in the payload. Their test shows that C&C communication is relative similar and can be used for detecting C&C communication of botnets in the cloud.

For spam-based botnets, Reference [190] proposes a method for the detection from spam mails received by those botnets. By looking at the mail header, the detection mechanism determines if the mail came from a botnet by looking at the sender's IP, the country of the domain name and the MX host of the sender. If the countries do not match, the sender is assumed to be part of a spam botnet. Reference [191] tackles botnet detection from a cyber-security standpoint, using a multiple detection mechanisms and aggregating the detection results in a central detection log for consideration. The used methods include honeypots, spam collection and recognition as well as high-level analysis based on known botnets. The techniques are based on a case study of the techniques applied at ACDC (Advanced Cyber Defense Centre) in Europe.

Reference [192] attempts to detect botnets by blocking botnet-infected hosts from sending mails. The proposed framework uses a whitelisting approach for running software within hosts, only allowing mails to be sent by authorised applications with a per-application encryption key. A process that sends mail without the authorisation key is flagged as a potential malware. Reference [193] uses a unique approach based on evidential reasoning detection botnets. In this approach, the actions of hosts are mined and reasoned to determine if the actions performed are within the expectations of the host. If not, the host may be detected as being part of a botnet.

#### 6.1.6. Mobile Botnets

For the purpose of clarification, Table 8 below details a number of papers that goes over mobile botnet based detection:

**Table 8.** Papers describing detection of botnets in mobile devices. Each column describes the overall technique, known advantages, disadvantages, detection rate and related papers, respectively.

| Technique | Advantage(s) | Disadvantage(s) | Detection Rate | Papers |
|---|---|---|---|---|
| Risk factor based on multi-category features | High accuracy in botnet apps.<br>Generates a pattern for Android botnet detection. | Only for static analysis.<br>No response mechanism. | Reference [194]: 93.1%. | [18,194] |
| Dynamic real-time analysis | N/A. | Limited analysis for risk factor. | N/A. | [19] |
| Machine learning | General high detection performance. | Bad detection performance when the bot coexists with other applications that communicate with many hosts. | Achieves 0.93 of the F-measure score by using graphlets of TCP and UDP with 10% of total traffic in 3-minute duration [195].<br>99.49% accuracy is achieved [196]. | [195,196] |
| Application monitoring | Mitigation: user warnings if something is suspicious. | SMS and social network applications are not monitored. | N/A. | [197] |

Another point of interest in botnet detection is the detection of smartphone-based botnets. Some papers, such as Abdullah and Saudi [18], propose assessing the potential risk of malicious apps by evaluating the API calls used by given apps. Apps shown to behave more like botnets are categorised as higher-risk and might potentially be blocked. Reference [18] also attempts to evaluate apps based on risk factors, weighing botnet-behaving apps as higher-risk compared to more benign apps. Reference [19] compares an app's permissions with a list of known harmful permissions and creates a threat level hierarchy on the basis of said permissions. Reference [194] extends the approach used in [19] by detecting botnets on the basis of both permissions and also the used API calls of each app. Reference [195] proposes the use of a graphlet-based machine learning algorithm on smartphone communication and then executing principal components analysis to identify P2P botnets on smartphones. Other approaches such as [196] run as an active agent on the smartphone OS to capture run-time data. The data is then labelled using a machine learning algorithm to determine whether or not an app acts like a botnet. Reference [197] instead asks the user to specify trusted apps and what permissions a given app should have according to the user. Any apps performing unauthorised or suspicious actions are flagged, and the user is informed. Periodic scans are performed to identify new threats and inform the user of unused apps.

#### 6.1.7. Vehicle Networks

With the development of autonomous vehicles, vehicular ad hoc networks (VANETs) have been designed to provide traffic safety by allowing the ad hoc transmission of safety information between vehicles. This additional communication makes VANETs a likely target for malicious attackers. Reference [198] introduces novel attack VANETs and propose a honeypot approach to notify nearby vehicles to ignore messages stemming from vehicles infected by botnets. The paper also proposes the use of localisation mechanisms to limit the exposure to far-away botnets. Reference [12] introduces *Shieldnet*, which employs a set of machine learning algorithms to detect the use of the *GHOST* [81] vehicular botnet. The algorithm detects suspicious activity by searching for outlier data within the Basic Safety Messages (BSM) fields of VANET broadcasts, also isolating known infected hosts using a reputation-based identification system.

#### 6.1.8. Social Network Botnets

Social network-based botnets (SnB) have become a major security issue in the past few years. Their incentives are based on sensitive information stealing, and perform complex communication procedures. Publicly available resources are highly vulnerable and provide an obfuscation layer in the C&C communication for botnets. The study [87] of this new malware method is essential to understand the actual challenges in detecting and mitigating social botnets. Social networks contain information such as sensitive and personal data of both registered and unregistered users. Moreover, it acts as a human-driven communication channel to share, talk and learn. This tool can be helpful in some respects, but it can also be destructive, e.g., propagating the influence of botnets [199].

To understand the behaviour of social network botnets, T. Yin and Y. Zhang and S. Li detail [60] the design and implementation of a Social Network-based botnet called DR-SNBot. The paper presents the necessary framework to deploy a C&C channel on the Sina blog website with a nickname generation algorithm and divide-and-conquer strategy. Compared to [60,166], it contains strong analysis on how to detect covert SnB in the real world. It is focused on the Stegobot and how to monitor host profile activity from a social network, and by extension, differentiate a normal profile from a Stegobot's one. Profiles are analysed by looking at their number of friends, likes and shares. A Stegobot has predictable patterns and communicates secret messages via carrier images, called "stego images", through the content sharing system from social networks. Their strategy is to study statistical correlation and build a classification algorithm using Machine Learning to identify malicious activities and suspicious accounts.

### *6.2. Mitigation Mechanisms*

After detecting a botnet and the threat they can represent, mitigation and countermeasures have to be deployed to limit the propagation of the botnet-enabling malware and protect the devices from being compromised. Mitigation mechanisms for botnets can be either reactive or proactive and can occur at different levels. The following section lists some of the mitigation strategies that can be employed when dealing with botnet and botnet-based attacks. These countermeasures can be found in Table 9 below.

**Table 9.** This table details the specific mitigation methods described in the following section, the advantages and disadvantages, as well as associated papers.

| Mitigation Mechanism | Advantages | Disadvantages | Papers |
|---|---|---|---|
| Best practices for end-users and organisations | Increases overall organisational security, considered best-practice, many well-known standards (i.e., ISO 27001). | Does not specifically target botnets, high user inconvenience cost. | [53,200] |
| Network-level blocking and packet analysis | Very high protection rate, many solutions and detection frameworks. | Very botnet-specific, can introduce additional latency at network edge. | [5,84,98,136,176,201–205] |
| Honeypots and botnet isolation | No effect on internal networks. Low cost. | Lower protection rate compared to network-level blocking, requires additional logically separated network. | [13,22,33,206–208] |
| Attacking P2P botnets | Helps mitigate botnet threat for others. Can target specific botnets. Hinders P2P advantages compared to centralised C&C models. | Only targets specific botnets (P2P-based). Low efficiency for organisations. | [209,210] |
| IoT-specific mitigation strategies | Low or offloaded compute resource cost. Some solutions provide general integrity for IoT-based networks. | Specific for IoT-based threats. Still few and untested options compared to network-level blocking. | [2,106,175,211,212] |
| Community-driven approaches | Potentially quicker adaption to newer botnets. Free and Open Source for organisations to use. | Dependent on community development. No de-facto standard decided yet. | [119,213] |
| Botnet mitigation with ethical issues (spreading anti-botnets, attacking suspected hosts) | Mitigates botnets for others. Slows botnet propagation. | Ethically questionable or illegal. Very specific per-botnet. | [49,82,214–217] |

#### 6.2.1. Best Practices for End-Users and Organisations

In general, following IT best practices is a good way to avoid the propagation and infection of botnets. An article of Justice news [53] coming from the Department of Justice (DOJ) of the United States announced “a multi-national effort to disrupt the Gameover Zeus Botnet”. The GameOver Zeus (GOZ) botnet is described as being capable of infecting victim computers to harvest credentials and banking information in order to gather millions of dollars from companies and customers. Therefore, a cybersecurity alert [200] at the National Cyber Awareness System has been released to explain how the botnet works and how attacks can be avoided. This assessment and mitigation document is written in collaboration with Department of Homeland Security (DHS), the DOJ and the Federal Bureau of Investigation (FBI). From this source, it is possible to get a grasp on the default methods of countermeasures used against every common botnet or malware:

- **Updating/changing passwords**: typically, botnets will try to access credentials from all connected devices and web accounts. The best way of protection is to follow the rules of ensuring high entropy of random password generation and execute frequent updates.

- **Updating devices**: infections are coming from unwanted vulnerabilities. Updating the operating system and the integrated software can help prevent devices from being compromised.
- **Updating/using anti-malware and anti-virus tools**: remediation tools and anti-viruses can erase malware infection and protect the device against new ones.
- **Being aware**: the hardest part to protect from is human behaviour. There ar multiple incentives, but botnets such as GOZ are mostly coming from spam and phishing messages, which can be avoided if the potential victim is aware of this potential threat source.

However, individual techniques are often not efficient enough to eradicate such threats. The Justice news article [53] explains the authorisation and capacity of redirecting requests made by the infected computers away from the malicious operators. With the evolution of the botnets detailed in this report, cyber defence needs to evolve and new mitigation techniques need to handle more complex attacks. Moreover, some of the newly described strategies only target specific types of botnets.

### 6.2.2. Network-Level Blocking and Packet Analysis

Within technical mitigation for botnet propagation, the use of network-level blocking is one of the most cited strategies. Many detection papers focus on network-level detection, which can be used by intrusion detection systems to block and contain botnets. In [202], an autonomous system (AS) is used to mitigate botnet threats. The AS stores a list of hosts' IP addresses and a threshold per host based on classification. Categories can be "Blacklist", "Whitelist", "Suspected Attacker" and "Possible Victim". AS are connected synchronously via the Ethereum blockchain. The threshold is monitored by every AS and refreshed after 20 s. Another way to ensure packets blocking is the software-defined networking approach [84,203]. The main purpose is to analyse the incoming packets rate at defined IoT switches to separate legitimate from malicious communication. Legitimate traffic is accepted, while malicious ones are blocked. Many mitigation strategies use a locally installed agent on host machines to block detected botnet traffic, informing the user of the infected nature of their machine [201]. Blocking can also be performed at the edge of the service provider but would face high implementation costs and requires some coordination across ISPs [204]. At the network level, removal of malware can be performed by agents installed locally or by the use of a continuous communication protocol with a master device. This validates of the integrity of local hosts and allows administrators to perform removal of botnet-enabling malware from hosts, either automatically or manually [5,176,205].

In [136], the authors propose a self-adaptive system for mitigation. In corporate area networks for instance, resilience can be ensured by using scenario-driven adaptive reconfiguration of networks. Scenarios are assessed and based on cluster analysis coming from previous botnet attacks. Moreover, the described system can apply more advanced actions such as reducing requests timeouts, decreasing allowed HTTP request size and blocking source hostname and IP addresses. In [98], a neural network-based system called BoNeSSy will notify the administrator if a threat is found and apply appropriate security actions such as blocking IP addresses or putting the system or suspicious network segment under surveillance. Many papers describe the detection of botnets using Machine Learning clustering via statistical behaviour correlation, but some of them are lacking of specific countermeasures description. Some characteristics can be countered by packets or IP addresses blocking.

### 6.2.3. Honeypots and Botnet Isolation

One of the most frequently described strategies [13] for mitigation is to isolate the botnet in order to perform information gathering and analysis of its behaviour and interaction via, for instance, honeypots [33,206] and honeynets [22]. From this information collection and assessment, it is possible to categorise the botnet based on behavioural characteristics and botnet structure. Organisations and researchers are producing and studying many methods of mitigation with various qualities and limitations. Honeypot behaviour has been shown to be detectable by intelligent botnets however. Although this is the case, the research and deployment of honeypots still has value for the scientific and industrial communities. The continued research in covert honeypots is therefore paramount to continue reaping the insights gained by using honeypots [207,208].

#### 6.2.4. Attacking P2P Botnets

Reference [209] proposes the use of poisoning of the routing table of P2P botnets as a potential mitigation method. By disrupting the majority of entries in the shared routing table of P2P botnets, it becomes possible to hinder some of the advantages that these types of botnets enjoy over the centralised model, such as resource efficiency and fault tolerance. Reference [210] also proposes disrupting P2P botnets but uses an optimised and tailored Sybil attack to infiltrate botnets and therefore mitigate them by disrupting or even taking them down from the inside. Placing Sybil nodes in the botnet shows that random placement is just as effective as informed placement due to the nature of P2P botnets. These nodes are able to disrupt communication between other nodes within the P2P botnet.

#### 6.2.5. Mitigation against IoT Attacks and Botnets

Learning from the Mirai botnet attack illustrates multiple general best practices, which can be used as a mitigation against IoT botnets. These methods include changing default credentials, closing unused service ports like telnet, detecting disabled watchdogs (Mirai specific) and the use of automated scripts to validate the implementation of the proposed mitigation [2,106]. Other proposed mitigation methods include switching from telnet to SSH (if possible) or changing the default service ports of services. Ensuring proper isolation of users and service account permissions and disabling any unencrypted communications (like HTTP) might mitigate some IoT botnet attacks [211]. Known ports vulnerable to attacks should also be continually monitored to quickly react to suspicious traffic [212]. Some local IoT agents have also been proposed to collectively mitigate the potential damage of DDoS attacks targeting local IoT installations [175].

#### 6.2.6. Community Driven Tools against Botnets

Reference [119] proposes the use of a community driven framework, *BotFlex*, to continually improve mitigation of botnets across the entire IT community. The approach attempts to standardise network-based intrusion detection systems with an extensible module system. Other researchers and corporations can contribute to the system with modules to improve upon *BotFlex*. In other community-driven approaches, Reference [213] proposes the use of a botnet defence description language to describe the tasks and information sharing primitives between devices handling botnet defence. Some community-driven efforts attempt to detect and prevent botnets by providing databases with known spam bots such as the The Spamhaus Project [218] and IBM X-Force exchange [219], where IT researchers can report suspected IP addresses and see a list of IP addresses along with a % indicator of how likely the IP is used for C&C. Furthermore Structured Threat Information eXpression (STIX) is used for exchanging cyber threat intelligence (CTI) as described in [220]. Dog et al. [221] examined the value of sharing IDS logs between enterprises and not just sharing IP addresses, domains and specific attacks. The study shows that intelligence sharing can provide good strategic threat information for enterprises.

#### 6.2.7. Botnet Mitigation with Potential Ethical Issues

Reference [214] discusses the ethical implications of fighting botnets with sinkholes. The information gathered by these sinkholes can be sold to government agencies, politicians, contractors and many more. This information includes geographical location of compromised hosts, operation system including version and the ability to target these already compromised hosts for future botnet or malware attacks. On the bright side, it could also help ISPs to provide their customers the service of malware protection. Another popular approach to mitigate botnets is to use their propagation mechanisms to propagate harmless versions of the given botnet. Actively ttacking the Mirai botnet and other IoT botnets to mitigate their threat has also been proposed [82]. Some researchers have tried to attack spam botnets to send unknowing users to more safe sites [215]. Another example, Reference [49], attempts to mitigate the Conficker botnet by spreading an anti-botnet, which blocks Conficker from executing and overtakes the propagation mechanism of Conficker to spread the anti-botnet instead. These approaches can be considered ethically problematic, as they intentionally spread (harmless as they might be) self-propagating malware [216,217].

## 7. Current Trends and Challenges

For the purpose of clarification, Table 10 below details a number of papers discussed in this section. The table denotes the overall topic, the overall trends within aforementioned topic, the relative interest for this specific trend, and a listing of all associated papers.

**Table 10.** Overview of papers discussing current trends and topics concerning botnets. The columns describes the trends of the overall associated area of interest, the detailed topics discussed in each paper, the relative interest amongst the associated trend and finally a listing of all the associated papers.

| Trend | Topics Within Trend | Relative Interest | Papers |
|---|---|---|---|
| Pervasive Computing | Spread of botnets in home appliances | 2 out 18 papers listed. | [31,201] |
| | Spread of botnets in mobile phones | 2 out 18 papers listed. | [222,223] |
| | Spread of botnets in (non)-autonomous vehicles. | 2 out 18 papers listed. | [81,198] |
| | Remotely disrupting the controls of an autonomous vehicle. | 1 out of 18 papers listed. | [81] |
| | Smartphones exploited via insufficient app certification process. | 2 out of 18 papers listed. | [37,197] |
| | Lack of restrictions hinders the process of avoiding botnet apps on mobile devices. | 2 out of 18 papers listed. | [207,211] |
| | Various proposals for IoT malware protection, both generalised and specialised. | 4 out of 18 papers listed. | [2,205,207,211] |
| | Usage of honeypots helps make more real-life like data for mitigation strategies. | 1 out of 18 papers listed. | [224] |
| | No standardised way to protect pervasive computing device hurts development of mitigation strategies. | 1 out of 18 papers listed. | [225] |
| | Best-practices in IT security yearns for standardising security in IoT and mobile devices. | 1 out of 18 papers listed. | [2] |
| Increasing complexity of botnets | Most firewalls and intrusion detection systems are not able to filter IPv6 traffic. | 2 out of 3 papers listed | [226,227] |
| | Modern botnets can circumvent traditional detection methods using encrypted channels for traditionally unencrypted traffic. | 1 out of 3 papers listed. | [228] |
| Social Botnets | Social botnets can be used for multiple purposes, including spam, C&C and falsifying/impersonating user behaviour. | 2 out of 4 papers listed. | [87,199] |
| | It is growing increasingly harder for users to discern between true and false information, benefiting botnets. | 1 out of 4 papers listed. | [166] |
| | New and more advanced counter measures are necessary to combat this new development of social botnets. | 1 out of 4 papers listed. | [1] |

**Table 10.** *Cont.*

| Trend | Topics Within Trend | Relative Interest | Papers |
|---|---|---|---|
| Machine learning and neural networks for botnet detection | New research on machine learning based detection schemes shows high rates of true positive detection of botnet behaviour. | 8 out of 9 papers listed. | [24,92,99,109,124,125,127,136] |
| | Discussion on the ability to train for zero-day vulnerabilities with custom created datasets. | 1 out of 9 papers listed. | [170] |
| Proactive botnet mitigation | Proactive botnet mitigation techniques show promising results. | 3 out of 6 papers listed. | [130,207,229] |
| | Proactive mitigation strategies and tools need to be developed for both the local and international stage. | 2 out of 6 papers listed. | [230,231] |
| | Users willing to pay for botnet prevention, but lack awareness. | 1 out of 6 papers listed. | [232] |
| Cloud-based botnets | Cloud services can be used for C&C communications between bots and bot masters, masquerading as benign user traffic. | 1 out of 3 papers listed. | [187] |
| | Cloud services offer options for researchers to create botnets without hassle. | 2 out of 3 papers listed. | [87,233] |

The current state of botnets and botnet research is consistently changing. As described in Section 5, on the topic of botnet evolution, botnets are constantly evolving. This section lists some of the general trends and challenges that have been identified during the reading for this paper.

### 7.1. The Continued Spread of Botnets within Pervasive Computing (VANETs, IoT and Mobile)

The most common trend and challenge within botnet research between 2013 and 2021 is the continued spread of botnets anchored in pervasive computing devices. With the increase in computational power within normally benign devices, such as home appliances [31,201], mobile phones [222,223] and (non)-autonomous vehicles [81,198], a higher potential of malicious activity within these devices becomes viable. As devices are allowed more computational headroom, botnets' ability to perform increasingly effective obfuscation techniques to mask their existence within pervasive devices grows ever more concerning.

Section 5 explains that the damage of botnets has mostly been within information channels, with attacks on the availability of computing system and acquisition of user credentials and national intelligence data being some of the primary targets of botnets. With pervasive computing, however, that threat of disruption transitions into the physical realm. Remotely disrupting the controls of an autonomous vehicle [81] can have potentially fatal results for the people within. Smart devices such as pacemakers and other computer-enabled medical devices may also provide a potentially fatal target for malicious actors or terrorists [234].

The mitigation of these attacks may vary greatly, depending on the specific scenario and device in question. Some devices, such as smartphones, are shown to be ripe for exploitation. An example of this is the app certification process, which has been shown to be insufficient [37,197] to prevent malicious apps from getting into various app stores. Furthermore, some device operating systems, like Android, do not place strict limitations on installing apps through unauthorised sources or package repositories. This considerably complicates the process of avoiding botnet apps on mobile. For IoT devices, attacks such as the Mirai botnet [207,211] shows the lack of basic security configuration and security investment within IoT development. Some solutions for IoT malware protection, both generalised and specialised, have been proposed within research though [2,205,207,211]. Attempts have also been made to make more real-life data based on honeypots available for researchers, in order to propose more IoT mitigation strategies [224]. However, so far no standardised ways to protect pervasive computing devices from botnets have been implemented. The lack of data sets for large IoT botnets in the wild is also seen as a challenge for the further development of mitigation against botnets targeting IoT installations [225]. With the differences in architecture and use-scenario of vehicles, IoT and mobile, a completely standardised approach across platforms might be a stretch. Some general best-practises within IT security, like avoiding default credentials, closing unused services and continuous validation of the platform, still apply for all pervasive computing devices [2].

### 7.2. Increasing Complexity of Botnets

Ravishankar expects future botnet threats to include encrypted communication, where a bot herder would encrypt the bot binary with a strong public/private key pair. Self-destruction mechanisms where the bot deletes registry files to try and enforce the user to reinstall the operation system and thereby get rid of most logged evidence, makes it hard for antivirus companies to analyse the botnets. Decentralised botnets such as P2P do not suffer from the vulnerability of a single point of failure, making the take down more complex. Tor-based onion routing can obfuscate communication to make eavesdropping and traffic analysis almost impossible. Tor can also be used for the bot herder to stay anonymous while setting up a new botnet. IPv6 can be misused to carry edited binary files and instructions to bots, malware tunnelling would also be possible in some situations, and lastly, most firewalls and intrusion detection systems are not yet able to filter IPv6 traffic [226,227]. Traditional detection methods by performing traffic analysis of DNS queries can be prevented by modern botnets utilising encrypted channels for traditionally unencrypted traffic [228].

### 7.3. Social Botnets

Social botnets are another challenge that have been under development for a while and are predicted to threaten online security and the integrity of information online. Social botnets specifically use social platforms for multiple purposes, including spam, C&C and falsifying or impersonating user behaviour [87] at an increasing rate [199]. This gives botnet developers more tools and ways to avoid typical detection mechanisms by communicating through seemingly benign social user accounts. Furthermore, as the amount of information processing of individuals increases and social botnets for the purpose of spam grow more advanced, it becomes increasingly unlikely for casual social network users to distinguish between true and false information [166]. Online social networks have recently stepped up by increasingly removing false social accounts. However, additional research is necessary to provide more generalised as well as specialised anti-measures for social botnets [1].

### 7.4. Machine Learning and Neural Networks for Botnet Detection

Detecting botnets using machine learning and neural networks has gained prominence amongst researchers and developers. Section 6 shows a clear majority of recent papers focusing on these techniques in order to achieve a high true positive rate of detection of botnet behaviour [24,92,99,109,124,125,127,136]. The additional advantages of potential real-time detection and mitigation and cloud-offloading for training/learning has allowed these techniques to establish themselves as the solution to botnet detection for the foreseeable future. Some parameters, such as the ability to train for zero-day botnet behaviour, are still a topic for discussion. Some papers, such as Zago et al.'s [170], have already tried to create data sets suitable to train machine learning algorithms for detecting certain patterns occurring within botnets. It is therefore expected that research will continue to attempt to improve upon the detection techniques based on machine learning and neural networks.

### 7.5. Proactive Botnet Mitigation

Most botnet detection mechanisms specified in Section 6 are *reactive* by nature. This allows botnets to flourish outside of properly protected and monitored networks. Because of the aforementioned reactive nature of both the detection and mitigation mechanisms, the proposed techniques only allow for protection at the local network level. This leaves many unprotected users vulnerable to botnet attacks, which may lead to technical and financial headaches for entities such as ISPs and the corporations supporting said users. Some proactive botnet mitigation techniques, such as honeypots and botnets overtaking other botnets show promising results, as documented in [130,207,229]. These new methods help raise user awareness while rendering other botnets harmless. An increased focus on proactive mitigation and detection strategies is necessary, not just to mitigate botnets at the local network level, but the international stage as well [230,231]. Research shows that users are willing to pay for services from their ISPs to prevent botnet attacks, but the lack of awareness nevertheless hurts the potential reach of such offerings [232].

### *7.6. Cloud-Based Botnets*

Finally, the normalisation of cloud-computing has allowed for greater computing power to both aid and combat botnets. This increase in computing resources and publicly available communication services, such as Google Cloud, has also been exploited by hackers to create botnets in the cloud. As cloud deployments are virtually instantaneous and on-demand, the cloud allows bot masters to dynamically scale the size of their botnets to match the needed computing power for attacks. Additionally, cloud services have been shown to be used for C&C communications between bots and bot masters, masquerading as benign user traffic [187]. Mitigation techniques based on IPs and locations are shown to be ineffective due to the relative in-deterministic nature of cloud deployment locality. The cloud can also be seen as a potentially attractive option for botnet research due to its low price, allowing researchers to instantly create botnets without having to make ethically questionable decisions such as infecting user computers in the wild [87,233].

## 8. Conclusions

This paper sought out to produce a novel systematic literature review detailing different subjects related to botnets, a growing subgroup of malware-enabled attacks. Botnets are widely used by malicious actors with various motivations and intentions, from simple denial-of-service attacks to advanced cyber espionage. The actors behind botnets therefore span a large range from security researchers spreading anti-botnets to foreign nations attempting to destabilise infrastructure.

The relatively simple structure and potential payoff of a successful attack has been a driving force in the evolution of botnets for decades. The adaptability of botnets are seen in their evolution towards modern platforms such as vehicles, smartphones and IoT devices. Modern botnets are still evolving rapidly, and more advanced counter-detection mechanisms and command-and-control channels are being introduced.

It has been shown that recent detection mechanisms based on machine learning and artificial neural networks provide very high rates of detecting botnet threats. Both these approaches provide additional accuracy to common network behaviour-based approaches. These detection techniques support traditional mitigation strategies such as security best practices and network-level blocking to reduce the risk and impact of botnet attacks.

In particular, the spread of pervasive computing paradigms such as the Internet of Things and vehicular networks provide a fertile ground for botnets to spread. Current trends point towards the increase of computing power within pervasive computing as an enabler for botnets to enable malware to spread. Other now-commonplace computing paradigms, such as cloud computing and interconnected social networks, have also seen an increase in interest as potential enablers of botnets.

**Author Contributions:** Conceptualization and methodology, all authors contributed equally; investigation and writing—original draft preparation, S.N.T.V., M.S., P.I.E.-H. and J.B.; writing—review and editing, all authors contributed equally; supervision, N.D.; project administration, N.D.; funding acquisition, N.D. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was partially funded by Industriens Fond (Danish Industry Foundation), project "CIDI–Cybersecure IoT in Danish Industry", grant number 2018-0197.

**Data Availability Statement:** Not applicable, the study does not report any data.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Silva, S.S.; Silva, R.M.; Pinto, R.C.; Salles, R.M. Botnets: A survey. *Comput. Netw.* **2013**, *57*, 378–403. Botnet Activity: Analysis, Detection and Shutdown. [CrossRef]
2. Margolis, J.; Oh, T.T.; Jadhav, S.; Kim, Y.H.; Kim, J.N. An In-Depth Analysis of the Mirai Botnet. In Proceedings of the 2017 International Conference on Software Security and Assurance (ICSSA), Altoona, PA, USA, 24–25 July 2017; pp. 6–12. [CrossRef]
3. Haria, S. The growth of the hide and seek botnet. *Netw. Secur.* **2019**, *2019*, 14–17. [CrossRef]
4. ENISA Threat Landscape 2020—Botnet. Available online: https://www.enisa.europa.eu/publications/enisa-threat-landscape-2020-botnet (accessed on 28 May 2021).
5. Hsiao, S.; Chen, Y.-N.; Sun, Y.S.; Chen, M.C. A cooperative botnet profiling and detection in virtualized environment. In Proceedings of the 2013 IEEE Conference on Communications and Network Security (CNS), National Harbor, MD, USA, 14–16 October 2013; pp. 154–162. [CrossRef]
6. European Union Agency Cybersecurity. Available online: https://www.enisa.europa.eu/topics/csirts-in-europe/glossary/botnets (accessed on 30 July 2020).
7. Zhang, W.; Wang, Y.J.; Wang, X.L. A Survey of Defense against P2P Botnets. In Proceedings of the 2014 IEEE 12th International Conference on Dependable, Autonomic and Secure Computing, Dalian, China, 24–27 August 2014; pp. 97–102. [CrossRef]
8. Ianelli, N.; Hackworth, A. Botnets as a Vehicle for Online Crime. *CERT Coord. Cent.* **2005**, *28*, 19–39. [CrossRef]
9. Etaher, N.; Weir, G.R.S.; Alazab, M. From ZeuS to Zitmo: Trends in Banking Malware. In Proceedings of the 2015 IEEE Trustcom/BigDataSE/ISPA, Helsinki, Finland, 20–22 August 2015; Volume 1, pp. 1386–1391. [CrossRef]
10. Elliott, C. Botnets: To what extent are they a threat to information security? *Inf. Secur. Tech. Rep.* **2010**, *15*, 79–103. Computer Crime—A 2011 Update. [CrossRef]
11. Eslahi, M.; Salleh, R.; Anuar, N.B. Bots and botnets: An overview of characteristics, detection and challenges. In Proceedings of the 2012 IEEE International Conference on Control System, Computing and Engineering, Penang, Malaysia, 23–25 November 2012; pp. 349–354. [CrossRef]
12. Garip, M.T.; Lin, J.; Reiher, P.; Gerla, M. SHIELDNET: An Adaptive Detection Mechanism against Vehicular Botnets in VANETs. In Proceedings of the 2019 IEEE Vehicular Networking Conference (VNC), Los Angeles, CA, USA, 4–6 December 2019; p. 9062790. [CrossRef]
13. Garg, S.; Sharma, R.M. Anatomy of botnet on application layer: Mechanism and mitigation. In Proceedings of the 2017 2nd International Conference for Convergence in Technology, I2CT 2017 Mumbai, India, 7–9 April 2017; pp. 1024–1029. [CrossRef]
14. Lange, T.; Kettani, H. On Security Threats of Botnets to Cyber Systems. In Proceedings of the 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 7–8 March 2019; pp. 176–183. [CrossRef]
15. García, S.; Zunino, A.; Campo, M. Survey on network-based botnet detection methods. *Secur. Commun. Netw.* **2014**, *7*, 878–903. [CrossRef]
16. Karim, A.; Salleh, R.; Shiraz, M.; Shah, S.; AWAN, I.; Anuar, N. Botnet detection techniques: Review, future trends and issues. *J. Zhejian Univ. Comput. Electron.* **2014**, *15*, 943–983. [CrossRef]
17. Khehra, G.; Sofat, S. Botnet Detection Techniques: A Review. In Proceedings of the 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 14–15 June 2018; pp. 1319–1326. [CrossRef]
18. Abdullah, Z.; Saudi, M. RAPID-Risk Assessment of Android Permission and Application Programming Interface (API) Call for Android Botnet. *Int. J. Emerg. Technol. Learn.* **2018**, *7*, 49–54. [CrossRef]
19. Kothari, S. Real Time Analysis of Android Applications by Calculating Risk Factor to Identify Botnet Attack. *Lect. Notes Electr. Eng.* **2020**, *570*, 55–62. [CrossRef]
20. Eslahi, M.; Salleh, R.; Anuar, N.B. MoBots: A new generation of botnets on mobile devices and networks. In Proceedings of the ISCAIE 2012—2012 IEEE Symposium on Computer Applications and Industrial Electronics, Kota Kinabalu, Malaysia, 3–4 December 2012; p. 6482109. [CrossRef]
21. Kaur, N.; Singh, M. Botnet and botnet detection techniques in cyber realm. In Proceedings of the 2016 International Conference on Inventive Computation Technologies (ICICT), Tamilnadu, India, 26–27 August 2016; Volume 3, pp. 1–7. [CrossRef]
22. Feily, M.; Shahrestani, A.; Ramadass, S. A Survey of Botnet and Botnet Detection. In Proceedings of the 2009 Third International Conference on Emerging Security Information, Systems and Technologies, Athens, Greece, 18–23 June 2009; pp. 268–273. [CrossRef]
23. Abdullah, R.; Abu, N.; Abdollah, M.; Muhamad Noh, Z.A. Understanding the Threats of Botnets Detection: A Wide Scale Survey. *Res. J. Inf. Technol.* **2014**, *6*, 135–153. [CrossRef]
24. Gaonkar, S.; Dessai, N.F.; Costa, J.; Borkar, A.; Aswale, S.; Shetgaonkar, P. A Survey on Botnet Detection Techniques. In Proceedings of the 2020 International Conference on Emerging Trends in Information Technology and Engineering (IC-ETITE), Vellore, India, 24–25 February 2020; pp. 1–6. [CrossRef]

25. Shetu, S.F.; Saifuzzaman, M.; Moon, N.N.; Nur, F.N. A Survey of Botnet in Cyber Security. In Proceedings of the 2019 2nd International Conference on Intelligent Communication and Computational Techniques (ICCT), Jaipur, India, 28–29 September 2019; pp. 174–177. [CrossRef]
26. Stevanovic, M.; Pedersen, J.M. An analysis of network traffic classification for botnet detection. In Proceedings of the 2015 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (Cybersa), London, UK, 8–9 June 2015; p. 8. [CrossRef]
27. Haddadi, F.; Le Cong, D.; Porter, L.; Zincir-Heywood, A.N. On the Effectiveness of Different Botnet Detection Approaches. In *Information Security Practice and Experience*; Lopez, J., Wu, Y., Eds.; Springer International Publishing: Cham, Switzerland, 2015; pp. 121–135.
28. Alazzam, H.; Alsmady, A.; Shorman, A.A. Supervised Detection of IoT Botnet Attacks. In Proceedings of the Second International Conference on Data Science, E-Learning and Information Systems, DATA '19, Dubai, United Arab Emirates, 2–5 December 2019; Association for Computing Machinery: New York, NY, USA, 2019; [CrossRef]
29. Falco, G.; Li, C.; Fedorov, P.; Caldera, C.; Arora, R.; Jackson, K. NeuroMesh: IoT security enabled by a blockchain powered botnet vaccine. *ACM Int. Conf. Proc. Ser.* **2019**, *148162*, 1–6. [CrossRef]
30. Hoque, N.; Bhattacharyya, D.K.; Kalita, J.K. Botnet in DDoS Attacks: Trends and Challenges. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2242–2270. [CrossRef]
31. Dange, S.; Chatterjee, M. IoT Botnet: The Largest Threat to the IoT Network. *Adv. Intell. Syst. Comput.* **2020**, *1049*, 137–157. [CrossRef]
32. Wazzan, M.; Algazzawi, D.; Bamasaq, O.; Albeshri, A.; Cheng, L. Internet of Things Botnet Detection Approaches: Analysis and Recommendations for Future Research. *Appl. Sci.* **2021**, *11*, 5713. [CrossRef]
33. Raghava, N.S.; Sahgal, D.; Chandna, S. Classification of Botnet Detection Based on Botnet Architechture. In Proceedings of the 2012 International Conference on Communication Systems and Network Technologies, Bangalore, India, 3–7 January 2012; pp. 569–572. [CrossRef]
34. Zhang, W.; Jin, C. The Research on Approaches for Botnet Detection. *Energy Procedia* **2011**, *13*, 9726–9732. [CrossRef]
35. Soe, Y.N.; Feng, Y.; Santosa, P.I.; Hartanto, R.; Sakurai, K. Machine learning-based IoT-botnet attack detection with sequential architecture. *Sensors* **2020**, *20*, 4372. [CrossRef] [PubMed]
36. Abraham, B.; Mandya, A.; Bapat, R.; Alali, F.; Brown, D.E.; Veeraraghavan, M. A Comparison of Machine Learning Approaches to Detect Botnet Traffic. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de janeiro, Brazil, 8–13 July 2018; pp. 1–8. [CrossRef]
37. Petersen, K.; Vakkalanka, S.; Kuzniarz, L. Guidelines for conducting systematic mapping studies in software engineering: An update. *Inf. Softw. Technol.* **2015**, *64*, 1–18. [CrossRef]
38. Wohlin, C. *Guidelines for Snowballing in Systematic Literature Studies and a Replication in Software Engineering*; EASE '14.; Association for Computing Machinery: New York, NY, USA, 2014. [CrossRef]
39. Petticrew, M.; Roberts, H. *Systematic Reviews in the Social Sciences: A Practical Guide*; John Wiley & Sons: Hoboken, NJ, USA, 2008. [CrossRef]
40. Digital Object Identifier FAQs. Available online: https://www.doi.org/faq.html (accessed on 14 December 2020).
41. Kolias, C.; Kambourakis, G.; Stavrou, A.; Voas, J. DDoS in the IoT: Mirai and other botnets. *Computer* **2017**, *50*, 80–84. [CrossRef]
42. Nazario, J. Politically motivated denial of service attacks. *Cryptol. Inf. Secur. Ser.* **2009**, *3*, 163–181. [CrossRef]
43. Sgouras, K.I.; Kyriakidis, A.N.; Labridis, D.P. Short-term risk assessment of botnet attacks on advanced metering infrastructure. *IET Cyber-Phys. Syst. Theory Appl.* **2017**, *2*, 143–151. [CrossRef]
44. Li, Z.; Liao, Q.; Blaich, A.; Striegel, A. Fighting botnets with economic uncertainty. *Secur. Commun. Netw.* **2011**, *4*, 1104–1113. [CrossRef]
45. Salamatian, S.; Huleihel, W.; Beirami, A.; Cohen, A.; Médard, M. Why Botnets Work: Distributed Brute-Force Attacks Need No Synchronization. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 2288–2299. [CrossRef]
46. Dev, J.A. Usage of botnets for high speed MD5 hash cracking. In Proceedings of the 2013 3rd International Conference on Innovative Computing Technology, Intech 2013, London, UK, 29–31 August 2013; p. 6653658. [CrossRef]
47. Bederna, Z.; Szadeczky, T. Cyber espionage through Botnets. *Secur. J.* **2020**, *33*, 43–62. [CrossRef]
48. Herwig, S.; Harvey, K.; Hughey, G.; Roberts, R.; Levin, D. Measurement and Analysis of Hajime, a Peer-to-peer IoT Botnet. In Proceedings of the Network and Distributed Systems Security (NDSS) Symposium, San Diego, CA, USA, 24–27 February 2019; p. 15. [CrossRef]
49. Xiang, C.; Lihua, Y.; Shuyuan, J.; Zhiyu, H.; Shuhao, L. Botnet spoofing: Fighting botnet with itself. *Secur. Commun. Netw.* **2015**, *8*, 80–89. [CrossRef]
50. Osagie, M.S.U.; Enagbonma, O.; Inyang, A.I. The Historical Perspective of Botnet Tools. *arXiv* **2019**, arXiv:1904.00948. [CrossRef].
51. Goodin, D. Record-Breaking DDoS Reportedly Delivered by >145 k hacked Cameras. Available online: arstechnica.com/information-technology/2016/09/botnet-of-145k-cameras-reportedly-deliver-internets-biggest-ddos-ever/ (accessed on 30 July 2021).
52. Fruhlinger, J. The Mirai Botnet Explained: How Teen Scammers and CCTV Cameras almost Brought down the Internet. Available online: csoonline.com/article/3258748/the-mirai-botnet-explained-how-teen-scammers-and-cctv-cameras-almost-brought-down-the-internet.html (accessed on 30 July 2021).

53. Office of Public Affairs (USA Department of Justice)—"U.S. Leads Multi-National Action against Gameover Zeus Botnet and Cryptolocker Ransomware, Charges Botnet Administrator". Available online: https://www.justice.gov/opa/pr/us-leads-multi-national-action-against-gameover-zeus-botnet-and-cryptolocker-ransomware (accessed on 30 July 2021).
54. Dabrowski, A.; Ullrich, J.; Weippl, E.R. Botnets causing blackouts: How coordinated load attacks can destabilize the power grid. *Elektrotechnik Und Informationstechnik* **2018**, *135*, 250–255. [CrossRef]
55. Zou, C.C.; Cunningham, R. Honeypot-aware advanced botnet construction and maintenance. *Proc. Int. Conf. Dependable Syst. Netw.* **2006**, *2006*, 1633509. [CrossRef]
56. Zeng, J.; Tang, W.; Liu, C.; Hu, J.; Peng, L. Efficient detect scheme of botnet command and control communication. *Commun. Comput. Inf. Sci.* **2012**, *307*, 576–581. [CrossRef]
57. Abu Rajab, M.; Zarfoss, J.; Monrose, F.; Terzis, A. A multifaceted approach to understanding the botnet phenomenon. In Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement, Rio de Janeriro, Brazil, 25–27 October 2006; pp. 41–52. [CrossRef]
58. Heron, S. Working the botnet: How dynamic DNS is revitalising the zombie army. *Netw. Secur.* **2007**, *2007*, 9–11. [CrossRef]
59. Liu, C.; Lu, W.; Zhang, Z.; Liao, P.; Cui, X. A recoverable hybrid C C botnet. In Proceedings of the 2011 6th International Conference on Malicious and Unwanted Software, Fajardo, PR, USA, 18–19 October 2011; pp. 110–118. [CrossRef]
60. Yin, T.; Zhang, Y.; Li, S. DR-SNBot: A Social Network-Based Botnet with Strong Destroy-Resistance. In Proceedings of the 2014 9th IEEE International Conference on Networking, Architecture, and Storage, Tianjin, China, 6–8 August 2014; pp. 191–199. [CrossRef]
61. Sood, A.K.; Zeadally, S.; Enbody, R.J. An Empirical Study of HTTP-based Financial Botnets. *IEEE Trans. Dependable Secur. Comput.* **2016**, *13*, 6991594. [CrossRef]
62. Wang, H.; Gong, Z. Collaboration-based botnet detection architecture. In Proceedings of the 2009 2nd International Conference on Intelligent Computing Technology and Automation, ICICTA 2009, Changsha, China, 10–11 October 2009; Volume 2, p. 5287910. [CrossRef]
63. Ogu, E.C.; Ojesanmi, O.A.; Awodele, O.; Kuyoro, S. A botnets circumspection: The current threat landscape, and what we know so far. *Information* **2019**, *10*, 337. [CrossRef]
64. Chen, Z.; Chen, C.; Wang, Q. Delay-Tolerant botnets. In Proceedings of the International Conference on Computer Communications and Networks, ICCCN, San Francisco, CA, USA, 3–6 August 2009; p. 5235321. [CrossRef]
65. Anagnostopoulos, M.; Kambourakis, G.; Gritzalis, S. New facets of mobile botnet: Architecture and evaluation. *Int. J. Inf. Secur.* **2016**, *15*, 455–473. [CrossRef]
66. Hamon, V. Android botnets for multi-targeted attacks. *J. Comput. Virol. Hacking Tech.* **2015**, *11*, 193–202. [CrossRef]
67. Mulliner, C.; Seifert, J.P. Rise of the iBots: Owning a telco network. In Proceedings of the 5th IEEE International Conference on Malicious and Unwanted Software, Malware 2010, Nancy, France, 19–20 October 2010; p. 5665790. [CrossRef]
68. Malatras, A.; Freyssinet, E.; Beslay, L. Mobile Botnets Taxonomy and Challenges. In Proceedings of the 2015 European Intelligence and Security Informatics Conference, EISIC 2015, Manchester, UK, 7–9 September 2015; p. 7379739. [CrossRef]
69. Rodriguez-Gomez, R.A.; Macia-Fernandez, G.; Garcia-Teodoro, P. Survey and taxonomy of botnet research through life-cycle. *ACM Comput. Surv.* **2013**, *45*, 2501659. [CrossRef]
70. Pieterse, H.; Olivier, M.S. Android botnets on the rise: Trends and characteristics. In Proceedings of the 2012 Information Security for South Africa—Proceedings of the ISSA 2012 Conference, Johannesburg, South Africa, 15–17 August 2012; p. 6320432. [CrossRef]
71. Chang, W.; Wang, A.; Mohaisen, A.; Chen, S. Characterizing botnets-as-a-service. In Proceedings of the Sigcomm 2014 ACM Conference on Special Interest Group on Data Communication, Chicago, IL, USA, 17–22 August 2014; Volume 44, pp. 585–586. [CrossRef]
72. Li, H.; Hu, G.; Yang, Y. Research on P2P botnet network behaviors and modeling. *Commun. Comput. Inf. Sci.* **2012**, *307*, 82–89. [CrossRef]
73. Aanjankumar, S.; Poonkuntran, S. An efficient soft computing approach for securing information over GAMEOVER Zeus Botnets with modified CPA algorithm. *Soft Comput.* **2020**, *24*, 16499–16507. [CrossRef]
74. Yan, G.; Ha, D.T.; Eidenbenz, S. AntBot: Anti-pollution peer-to-peer botnets. *Comput. Netw.* **2011**, *55*, 1941–1956. [CrossRef]
75. Andriesse, D.; Rossow, C.; Stone-Gross, B.; Plohmann, D.; Bos, H. Highly resilient peer-to-peer botnets are here: An analysis of Gameover Zeus. In Proceedings of the 2013 8th International Conference on Malicious and Unwanted Software: "The Americas", Malware 2013, Fajardo, PR, USA, 22–24 October 2013; p. 6703693. [CrossRef]
76. Zhuang, D.; Morris Chang, J. Enhanced PeerHunter: Detecting Peer-To-Peer Botnets Through Network-Flow Level Community Behavior Analysis. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 8536452. [CrossRef]
77. Rossow, C.; Andriesse, D.; Werner, T.; Stone-Gross, B.; Plohmann, D.; Dietrich, C.J.; Bos, H. SoK: P2PWNED—Modeling and evaluating the resilience of peer-to-peer botnets. In Proceedings of the IEEE Symposium on Security and Privacy, Berkeley, CA, USA, 19–22 May 2013; p. 6547104. [CrossRef]
78. Wang, T.; Wang, H.; Liu, B.; Shi, P. What is the pattern of a botnet? In Proceedings of the 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, Trustcom 2013, Melbourne, Australia, 16–18 July 2013; p. 6680849. [CrossRef]

79. Perrotta, R.; Hao, F. Botnet in the browser: Understanding threats caused by malicious browser extensions. *IEEE Secur. Priv.* **2018**, *16*, 8425617. [CrossRef]
80. Boshmaf, Y.; Muslukhov, I.; Beznosov, K.; Ripeanu, M. Design and analysis of a social botnet. *Comput. Netw.* **2013**, *57*, 556–578. [CrossRef]
81. Garip, M.T.; Reiher, P.; Gerla, M. Ghost: Concealing vehicular botnet communication in the VANET control channel. In Proceedings of the 2016 International Wireless Communications and Mobile Computing Conference, IWCMC 2016, Paphos, Cyprus, 5–9 September 2016; p. 7577024. [CrossRef]
82. Yamaguchi, S. Botnet defense system: Concept, design, and basic strategy. *Information* **2020**, *11*, 516. [CrossRef]
83. Bertino, E.; Islam, N. Botnets and Internet of Things Security. *Computer* **2017**, *50*, 7842850. [CrossRef]
84. Mendes, L.D.; Aloi, J.; Pimenta, T.C. Analysis of IoT botnet architectures and recent defense proposals. *Proc. Int. Conf. Microelectron. ICM* **2019**, *2019*, 9021715. [CrossRef]
85. Kudo, T.; Kimura, T.; Inoue, Y.; Aman, H.; Hirata, K. Behavior analysis of self-evolving botnets. In Proceedings of the IEEE CITS 2016—2016 International Conference on Computer, Information and Telecommunication Systems, Kunming, China, 6–8 July 2016; p. 7546428. [CrossRef]
86. Bock, L.; Alexopoulos, N.; Saracoglu, E.; Muhlhauser, M.; Vasilomanolakis, E. Assessing the Threat of Blockchain-based Botnets. *Ecrime Res. Summit Ecrime* **2019**, *2019*, 9037600. [CrossRef]
87. Yin, J.; Lv, H.; Zhang, F.; Tian, Z.; Cui, X. Study on advanced botnet based on publicly available resources. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer: Cham, Switzerland, 2018; Volume 11149, pp. 57–74. [CrossRef]
88. Hua, J.; Sakurai, K. A SMS-Based Mobile Botnet Using Flooding Algorithm. In *Information Security Theory and Practice. Security and Privacy of Mobile Devices in Wireless Communication*; Ardagna, C.A., Zhou, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; pp. 264–279.
89. Geng, G.; Xu, G.; Zhang, M.; Yang, Y.; Yang, G. An improved SMS based heterogeneous mobile botnet model. In Proceedings of the 2011 IEEE International Conference on Information and Automation, ICIA 2011, Shenzhen, China, 6–8 June 2011; p. 5948987. [CrossRef]
90. Lee, H.; Kang, T.; Lee, S.; Kim, J.; Kim, Y. Punobot: Mobile Botnet Using Push Notification Service in Android. In *Information Security Applications*; Kim, Y., Lee, H., Perrig, A., Eds.; Springer International Publishing: Cham, Switzerland, 2014; pp. 124–137.
91. Porras, P.; Saidi, H.; Yegneswaran, V. An Analysis of the iKee.B iPhone Botnet. In *Security and Privacy in Mobile Information and Communication*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 141–152.
92. Li, X.G.; Wang, J.F. Traffic detection of transmission of botnet threat using BP neural network. *Neural Netw. World* **2018**, *28*, 511–522. [CrossRef]
93. Nguyen, H.T.; Ngo, Q.D.; Le, V.H. A novel graph-based approach for IoT botnet detection. *Int. J. Inf. Secur.* **2020**, *19*, 567–577. [CrossRef]
94. Pei, Z.; Gan, G. Research on p2p botnet traffic identification technology based on neural network. *IOP Conf. Ser. Earth Environ. Sci.* **2020**, *428*, 012011. [CrossRef]
95. Taheri, S.; Salem, M.; Yuan, J.S. Leveraging image representation of network traffic data and transfer learning in botnet detection. *Big Data Cogn. Comput.* **2018**, *2*, 37. [CrossRef]
96. Jung, W.; Zhao, H.; Sun, M.; Zhou, G. IoT botnet detection via power consumption modeling. *Smart Health* **2020**, *15*, 100103. [CrossRef]
97. Kim, J.; Shim, M.; Hong, S.; Shin, Y.; Choi, E. Intelligent detection of iot botnets using machine learning and deep learning. *Appl. Sci.* **2020**, *10*, 7009. [CrossRef]
98. Nogueira, A.; Salvador, P.; Blessa, F. A botnet detection system based on neural networks. In Proceedings of the 5th International Conference on Digital Telecommunications, ICDT 2010, Athens, Greece, 13–19 June 2010; p. 5532380. [CrossRef]
99. Javed, Y.; Rajabi, N. Multi-Layer Perceptron Artificial Neural Network Based IoT Botnet Traffic Classification. *Adv. Intell. Syst. Comput.* **2020**, *1069*, 973–984. [CrossRef]
100. Zeidanloo, H.R.; Hosseinpour, F.; Borazjani, P.N. Botnet detection based on common network behaviors by utilizing Artificial Immune System(AIS). In Proceedings of the ICSTE 2010—2010 2nd International Conference on Software Technology and Engineering, San Juan, PR, USA, 3–5 October 2010; Volume 1, p. 5608967. [CrossRef]
101. Sriram, S.; Vinayakumar, R.; Alazab, M.; Soman, K.P. Network flow based IoT botnet attack detection using deep learning. In Proceedings of the IEEE Infocom 2020—IEEE Conference on Computer Communications Workshops, Infocom WKSHPS 2020, Toronto, ON, Canada, 6–9 July 2020; p. 9162668. [CrossRef]
102. Jithu, P.; Shareena, J.; Ramdas, A.; Haripriya, A.P. Intrusion Detection System for IOT Botnet Attacks Using Deep Learning. *SN Comput. Sci.* **2021**, *2*, 1–8. [CrossRef]
103. Alharbi, A.; Alosaimi, W.; Alyami, H.; Rauf, H.T.; Damasevicius, R. Botnet Attack Detection Using Local Global Best Bat Algorithm for Industrial Internet of Things. *Electronics* **2021**, *10*, 1341. [CrossRef]
104. Kanehara, H.; Takahashi, T.; Murakami, Y.; Inoue, D.; Shimamura, J.; Murata, N. Real-time botnet detection using nonnegative tucker decomposition. *Proc. ACM Symp. Appl. Comput.* **2019**, *147772*, 1337–1344. [CrossRef]
105. Bansal, A.; Mahapatra, S. A Comparative Analysis of Machine Learning Techniques for Botnet Detection. In Proceedings of the 10th International Conference on Security of Information and Networks, Jaipur, India, 13–15 October 2017; pp. 91–100. [CrossRef]

106. Eustis, A.G. The Mirai Botnet and the Importance of IoT Device Security. In Proceedings of the 16th International Conference on Information Technology-New Generations (ITNG 2019), Las Vegas, NV, USA, 1–3 April 2019; Latifi, S., Ed.; Springer International Publishing: Cham, Switzerland, 2019; pp. 85–89.
107. Ribeiro, G.H.; De Faria Paiva, E.R.; Miani, R.S. A comparison of stream mining algorithms on botnet detection. In Proceedings of the 15th International Conference on Availability, Reliability and Security, Dublin, Ireland, 25–28 August 2020; pp. 1–10. [CrossRef]
108. Chu, Z.; Han, Y.; Zhao, K. Botnet Vulnerability Intelligence Clustering Classification Mining and Countermeasure Algorithm Based on Machine Learning. *IEEE Access* **2019**, *7*, 8935236. [CrossRef]
109. Tuan, T.A.; Long, H.V.; Son, L.H.; Kumar, R.; Priyadarshini, I.; Son, N.T.K. Performance evaluation of Botnet DDoS attack detection using machine learning. *Evol. Intell.* **2020**, *13*, 283–294. [CrossRef]
110. Indre, I.; Lemnaru, C. Detection and prevention system against cyber attacks and botnet malware for information systems and Internet of Things. In Proceedings of the 2016 IEEE 12th International Conference on Intelligent Computer Communication and Processing, ICCP 2016, Cluj-Napoca, Romania, 8–10 September 2016; p. 7737142. [CrossRef]
111. Park, Y.; Kengalahalli, N.V.; Chang, S.Y. Distributed Security Network Functions against Botnet Attacks in Software-defined Networks. In Proceedings of the 2018 IEEE Conference on Network Function Virtualization and Software Defined Networks, NFV-SDN 2018, Verona, Italy, 27–29 November 2018; p. 8725657. [CrossRef]
112. Lu, W.; Tavallaee, M.; Ghorbani, A.A. Automatic Discovery of Botnet Communities on Large-Scale Communication Networks. In Proceedings of the 4th International Symposium on Information, Computer, and Communications Security, ASIACCS ’09, Auckland, New Zealand, 9–12 July 2009; Association for Computing Machinery: New York, NY, USA, 2009; pp. 1–10. [CrossRef]
113. Goyal, M.; Sahoo, I.; Geethakumari, G. HTTP Botnet Detection in IOT Devices using Network Traffic Analysis. In Proceedings of the 2019 International Conference on Recent Advances in Energy-Efficient Computing and Communication, ICRAECC 2019, Nagercoil, India, 7–20 March 2019; p. 8995160. [CrossRef]
114. Heydari, B.; Yajam, H.; Akhaee, M.A.; Salehkalaibar, S. Utilizing Features of Aggregated Flows to Identify Botnet Network Traffic. In Proceedings of the 2017 14th International ISC (Iranian Society of Cryptology) Conference on Information Security and Cryptology, ISCISC 2017, Shiraz, Iran, 6–7 September 2017; p. 8488370. [CrossRef]
115. Haddadi, F.; Morgan, J.; Filho, E.G.; Zincir-Heywood, A.N. Botnet behaviour analysis using IP flows: With http filters using classifiers. In Proceedings of the 2014 IEEE 28th International Conference on Advanced Information Networking and Applications Workshops, IEEE Waina 2014, Victoria, BC, Canada, 13–16 May 2014; p. 6844605. [CrossRef]
116. Yong, W.; Tefera, S.H.; Beshah, Y.K. Understanding botnet: From mathematical modelling to integrated detection and mitigation framework. In Proceedings of the 13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/distributed Computing, SNPD 2012, Kyoto, Japan, 8–10 August 2012; p. 6299259. [CrossRef]
117. AsSadhan, B.; Bashaiwth, A.; Al-Muhtadi, J.; Alshebeili, S. Analysis of P2P, IRC and HTTP traffic for botnets detection. *Peer-to-Peer Netw. Appl.* **2018**, *11*, 848–861. [CrossRef]
118. Zand, A.; Vigna, G.; Yan, X.; Kruegel, C. Extracting probable command and control signatures for detecting botnets. In Proceedings of the 29th Annual ACM Symposium on Applied Computing, Gyeongju, Korea, 24–28 March; pp. 1657–1662. [CrossRef]
119. Khattak, S.; Ahmed, Z.; Syed, A.A.; Khayam, S.A. BotFlex: A community-driven tool for botnet detection. *J. Netw. Comput. Appl.* **2015**, *58*, 144–154. [CrossRef]
120. Richer, T.J. Entropy-based detection of botnet command and control. In Proceedings of the Australasian Computer Science Week Multiconference, Geelong, Australia, 31 January–3 February 2017; p. a75. [CrossRef]
121. AsSadhan, B.; Moura, J.M. An efficient method to detect periodic behavior in botnet traffic by analyzing control plane traffic. *J. Adv. Res.* **2014**, *5*, 435–448. [CrossRef]
122. Lagraa, S.; François, J.; Lahmadi, A.; Miner, M.; Hammerschmidt, C.; State, R. BotGM: Unsupervised graph mining to detect botnets in traffic flows. In Proceedings of the 2017 1st Cyber Security in Networking Conference, CSNET 2017, Rio de Janeiro, Brazil, 18–20 October 2017; pp. 1–8. [CrossRef]
123. Sousa, R.; Rodrigues, N.; Salvador, P.; Nogueira, A. Analyzing the Behavior of Top Spam Botnets. In Proceedings of the 2012 IEEE International Conference on Communications (ICC), Ottawa, ON, Canada, 10–15 June 2012; pp. 6540–6544. [CrossRef]
124. Kozik, R.; Choraś, M. Pattern Extraction Algorithm for NetFlow-Based Botnet Activities Detection. *Secur. Commun. Netw.* **2017**, *2017*, 6047053. [CrossRef]
125. Chen, R.; Niu, W.; Zhang, X.; Zhuo, Z.; Lv, F. An Effective Conversation-Based Botnet Detection Method. *Math. Probl. Eng.* **2017**, *2017*, 4934082. [CrossRef]
126. Pandey, A.; Thaseen, S.; Aswani Kumar, C.; Li, G. Identification of botnet attacks using hybrid machine learning models. *Adv. Intell. Syst. Comput.* **2021**, *1179*, 249–257. [CrossRef]
127. Kirubavathi, G.; Anitha, R. Botnet detection via mining of traffic flow characteristics. *Comput. Electr. Eng.* **2016**, *50*, 91–101. [CrossRef]
128. Li, S.H.; Kao, Y.C.; Zhang, Z.C.; Chuang, Y.P.; Yen, D.C. A network behavior-based botnet detection mechanism using PSO and K-means. *ACM Trans. Manag. Inf. Syst.* **2015**, *6*, 3. [CrossRef]
129. Su, S.C.; Chen, Y.R.; Tsai, S.C.; Lin, Y.B. Detecting P2P Botnet in Software Defined Networks. *Secur. Commun. Netw.* **2018**, *2018*, 4723862. [CrossRef]

130. Vishwakarma, R.; Jain, A.K. A Honeypot with Machine Learning based Detection Framework for defending IoT based Botnet DDoS Attacks. In Proceedings of the 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 23–25 April 2019; pp. 1019–1024. [CrossRef]
131. Al-Hakbani, M.M.; Dahshan, M.H. Avoiding honeypot detection in peer-to-peer botnets. In Proceedings of the ICETECH 2015—2015 IEEE International Conference on Engineering and Technology, Coimbatore, India, 20 March 2015; p. 7275017. [CrossRef]
132. Dwyer, O.P.; Marnerides, A.K.; Giotsas, V.; Mursch, T. Profiling iot-based botnet traffic using DNS. In Proceedings of the 2019 IEEE Global Communications Conference, Globecom 2019, Waikoloa, HI, USA, 9–13 December 2019; p. 9014300. [CrossRef]
133. Wang, K.; Huang, C.Y.; Tsai, L.Y.; Lin, Y.D. Behavior-based botnet detection in parallel. *Secur. Commun. Netw.* **2014**, *7*, 1849–1859. [CrossRef]
134. Bahsi, H.; Nomm, S.; La Torre, F.B. Dimensionality Reduction for Machine Learning Based IoT Botnet Detection. In Proceedings of the 2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV), Singapore, 18–21 November 2018; pp. 1857–1862. [CrossRef]
135. Beigi, E.B.; Jazi, H.H.; Stakhanova, N.; Ghorbani, A.A. Towards effective feature selection in machine learning-based botnet detection approaches. In Proceedings of the 2014 IEEE Conference on Communications and Network Security, CNS 2014, San Francisco, CA, USA, 29–31 October 2014; p. 6997492. [CrossRef]
136. Lysenko, S.; Savenko, O.; Bobrovnikova, K.; Kryshchuk, A. Self-adaptive system for the corporate area network resilience in the presence of botnet cyberattacks. *Commun. Comput. Inf. Sci.* **2018**, *860*, 385–401. [CrossRef]
137. Hung, C.L.; Wang, H.H. Parallel botnet detection system by using GPU. In Proceedings of the 2014 IEEE/ACIS 13th International Conference on Computer and Information Science, ICIS 2014, Taiyuan, China, 4–6 June 2014; p. 6912109. [CrossRef]
138. Blaise, A.; Bouet, M.; Conan, V.; Secci, S. Botnet Fingerprinting: A Frequency Distributions Scheme for Lightweight Bot Detection. *IEEE Trans. Netw. Serv. Manag.* **2020**, *17*, 9097931. [CrossRef]
139. Lin, S.C.; Chen, P.S.; Chang, C.C. A novel method of mining network flow to detect P2P botnets. *Peer-to-Peer Netw. Appl.* **2014**, *7*, 645–654. [CrossRef]
140. Lee, Y.C.; Tseng, C.M.; Liu, T.J. A HTTP botnet detection system based on ranking mechanism. In Proceedings of the 2017 12th International Conference on Digital Information Management, ICDIM 2017, Fukuoka, Japan, 12–14 September2017; pp. 115–120. [CrossRef]
141. Mai, L.; Noh, D.K. Cluster Ensemble with Link-Based Approach for Botnet Detection. *J. Netw. Syst. Manag.* **2018**, *26*, 616–639. [CrossRef]
142. Lu, W.; Ghorbani, A.A. Botnets Detection Based on IRC-Community. In Proceedings of the IEEE GLOBECOM 2008—2008 IEEE Global Telecommunications Conference, New Orleans, LA, USA, 30 November–4 December 2008; pp. 1–5. [CrossRef]
143. Hoang, X.D.; Nguyen, Q.C. Botnet detection based on machine learning techniques using DNS query data. *Future Internet* **2018**, *10*, 43. [CrossRef]
144. Wang, Z.; Qin, M.; Chen, M.; Jia, C.; Ma, Y. A learning evasive email-based P2P-Like botnet. *China Commun.* **2018**, *15*, 15–24. [CrossRef]
145. Rezaei, A. Using Ensemble Learning Technique for Detecting Botnet on IoT. *SN Comput. Sci.* **2021**, *2*, 1–14. [CrossRef]
146. Lee, S.; Abdullah, A.; Jhanjhi, N.Z.; Kok, S.H. Honeypot Coupled Machine Learning Model for Botnet Detection and Classification in IoT Smart Factory—An Investigation. *MATEC Web Conf.* **2021**, *335*, 04003. [CrossRef]
147. Ibrahim, W.N.H.; Anuar, S.; Selamat, A.; Krejcar, O.; Gonzalez Crespo, R.; Herrera-Viedma, E.; Fujita, H. Multilayer Framework for Botnet Detection Using Machine Learning Algorithms. *IEEE Access* **2021**, *9*, 9359784. [CrossRef]
148. Hao, S.; Liu, D.; Baldi, S.; Yu, W. Unsupervised detection of botnet activities using frequent pattern tree mining. *Complex Intell. Syst.* **2021**, 1–9. [CrossRef]
149. Asadi, M. Detecting IoT botnets based on the combination of cooperative game theory with deep and machine learning approaches. *J. Ambient. Intell. Humaniz. Comput.* **2021**, 1–15. [CrossRef]
150. Bhatt, P.; Thakker, B. A Novel Forecastive Anomaly Based Botnet Revelation Framework for Competing Concerns in Internet of Things. *J. Appl. Secur. Res.* **2021**, *16*, 258–278. [CrossRef]
151. Soleymani, A.; Arabgol, F. A Novel Approach for Detecting DGA-Based Botnets in DNS Queries Using Machine Learning Techniques. *J. Comput. Netw. Commun.* **2021**, *2021*, 4767388. [CrossRef]
152. Panda, M.; Allah A. Mousa, A.; Ella Hassanien, A. Developing an Efficient Feature Engineering and Machine Learning Model for Detecting IoT-Botnet Cyber Attacks. *IEEE Access* **2021**, *9*, 91038–91052. [CrossRef]
153. Kwon, J.; Lee, J.; Lee, H.; Perrig, A. PsyBoG: A scalable botnet detection method for large-scale DNS traffic. *Comput. Netw.* **2016**, *97*, 48–73. [CrossRef]
154. Wang, T.S.; Lin, H.T.; Cheng, W.T.; Chen, C.Y. DBod: Clustering and detecting DGA-based botnets using DNS traffic analysis. *Comput. Secur.* **2017**, *64*, 1–15. [CrossRef]
155. Zhao, D.; Traore, I.; Sayed, B.; Lu, W.; Saad, S.; Ghorbani, A.; Garant, D. Botnet detection based on traffic behavior analysis and flow intervals. *Comput. Secur.* **2013**, *39*, 2–16. [CrossRef]
156. Ichise, H.; Jin, Y.; Iida, K. Analysis of via-resolver DNS TXT queries and detection possibility of botnet communications. In Proceedings of the IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing, Victoria, BC, Canada, 24–26 August 2015; p. 7334837. [CrossRef]

157. Jin, Y.; Ichise, H.; Iida, K. Design of Detecting Botnet Communication by Monitoring Direct Outbound DNS Queries. In Proceedings of the 2nd IEEE International Conference on Cyber Security and Cloud Computing, Cscloud 2015—IEEE International Symposium of Smart Cloud, IEEE SSC 2015, New York, NY, USA, 3–5 November 2015; p. 7371456. [CrossRef]
158. Nguyen, T.D.; Dung, T.C.; Nguyen, L.G. DGA botnet detection using collaborative filtering and density-based clustering. In Proceedings of the Sixth International Symposium on Information and Communication Technology, Hue, Vietnam, 3–4 December 2015; pp. 203–209. [CrossRef]
159. Mohd Safar, N.; Abdullah, N.; Kamaludin, H.; Abd Ishak, S.; Isa, M. Characterising and detection of botnet in P2P network for UDP protocol. *Indones. J. Electr. Eng. Comput. Sci.* **2020**, *18*, 1584–1595. [CrossRef]
160. Tsai, M.H.; Chang, K.C.; Lin, C.C.; Mao, C.H.; Lee, H.M. C&C tracer: Botnet command and control behavior tracing. In Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Anchorage, AK, USA, 9–12 October 2011; p. 6083942. [CrossRef]
161. Lysenko, S.; Bobrovnikova, K.; Savenko, O.; Kryshchuk, A. BotGRABBER: SVM-Based Self-Adaptive System for the Network Resilience against the Botnets Cyberattacks. *Commun. Comput. Inf. Sci.* **2019**, *1039*, 127–143. [CrossRef]
162. Tong, V.; Nguyen, G. A method for detecting DGA botnet based on semantic and cluster analysis. In Proceedings of the Seventh Symposium on Information and Communication Technology, Ho Chi Minh, Vietnam, 8–9 December 2016; pp. 272–277. [CrossRef]
163. Kelley, T.; Furey, E. Getting Prepared for the Next Botnet Attack: Detecting Algorithmically Generated Domains in Botnet Command and Control. In Proceedings of the 2018 29th Irish Signals and Systems Conference (ISSC), Belfast, UK, 21–22 June 2018; pp. 1–6. [CrossRef]
164. Vishvakarma, D.K.; Bhatia, A.; Riha, Z. Detection of Algorithmically Generated Domain Names in Botnets. In *Advanced Information Networking and Applications*; Barolli, L., Takizawa, M., Xhafa, F., Enokido, T., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 1279–1290.
165. Truong, D.T.; Cheng, G. Detecting domain-flux botnet based on DNS traffic features in managed network. *Secur. Commun. Netw.* **2016**, *9*, 2338–2347. [CrossRef]
166. Natarajan, V.; Sheen, S.; Anitha, R. Multilevel analysis to detect covert social botnet in multimedia social networks. *Comput. J.* **2015**, *58*, 679–687. [CrossRef]
167. Alhomoud, A.; Awan, I.; Pagna Disso, J.F.; Younas, M. A next-generation approach to combating botnets. *Computer* **2013**, *46*, 6459493. [CrossRef]
168. Sharafaldin, I.; Gharib, A.; Lashkari, A.H.; Ghorbani, A.A. BotViz: A memory forensic-based botnet detection and visualization approach. In Proceedings of the 2017 International Carnahan Conference on Security Technology (ICCST), Madrid, Spain, 23–26 October 2017; pp. 1–8. [CrossRef]
169. Virustotal.com. 2020. Available online: Virustotal.com (accessed on 18 December 2020).
170. Zago, M.; Gil Perez, M.; Martinez Perez, G. UMUDGA: A dataset for profiling algorithmically generated domain names in botnet detection. *Data Brief* **2020**, *30*, 105400. [CrossRef]
171. k. Idriss, H. Mirai Botnet In Lebanon. In Proceedings of the 2020 8th International Symposium on Digital Forensics and Security (ISDFS), Beirut, Lebanon, 1–2 June 2020; pp. 1–6. [CrossRef]
172. Balasubramanian, Y.; Baggam, D.S.; Venkatraman, S.; Ramaswamy, V. Quantum IDS for mitigation of DDoS attacks by mirai botnets. *Commun. Comput. Inf. Sci.* **2018**, *828*, 488–501. [CrossRef]
173. Tzagkarakis, C.; Petroulakis, N.; Ioannidis, S. Botnet Attack Detection at the IoT Edge Based on Sparse Representation. In Proceedings of the 2019 Global IoT Summit (GIoTS), Aarhus, Denmark, 17–21 June 2019; pp. 1–6. [CrossRef]
174. Prokofiev, A.O.; Smirnova, Y.S.; Surov, V.A. A method to detect Internet of Things botnets. In Proceedings of the 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering, Elconrus 2018, Moscow, Russia; St. Petersburg, Russia, 29 January–1 February 2018; pp. 105–108. [CrossRef]
175. Giachoudis, N.; Damiris, G.P.; Theodoridis, G.; Spathoulas, G. Collaborative agent-based detection of DDoS IoT botnets. In Proceedings of the 15th Annual International Conference on Distributed Computing in Sensor Systems, DCOSS 2019, Santorini Island, Greece, 29–31 May 2019; p. 8804480. [CrossRef]
176. Spathoulas, G.; Giachoudis, N.; Damiris, G.P.; Theodoridis, G. Collaborative blockchain-based detection of distributed denial of service attacks based on internet of things botnets. *Future Internet* **2019**, *11*, 226. [CrossRef]
177. Cui, P.; Guin, U. Countering Botnet of Things using Blockchain-Based Authenticity Framework. In Proceedings of the 2019 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), Miami, FL, USA, 15–17 July 2019; p. 8839425. [CrossRef]
178. Zareh, A.; Shahriari, H.R. BotcoinTrap: Detection of Bitcoin Miner Botnet Using Host Based Approach. In Proceedings of the 2018 15th International ISC (Iranian Society of Cryptology) Conference on Information Security and Cryptology, ISCISC 2018, Tehran, Iran, 28–29 August 2018; p. 8546867. [CrossRef]
179. Zhuang, D.; Chang, J.M. PeerHunter: Detecting peer-to-peer botnets through community behavior analysis. In Proceedings of the 2017 IEEE Conference on Dependable and Secure Computing, Taipei, Taiwan, 7–10 August 2017; p. 8073832. [CrossRef]
180. Priyanka; Dave, M. PeerFox: Detecting parasite P2P botnets in their waiting stage. In Proceedings of the 2015 International Conference on Signal Processing, Computing and Control, ISPCC 2015, Solan, India, 24–26 September 2015; p. 7375054. [CrossRef]
181. Obeidat, A.A.; Al-Kofahi, M.M.; Bawaneh, M.J.; Hanandeh, E.S. A novel botnet detection system for P2P networks. *J. Comput. Sci.* **2017**, *13*, 329–336. [CrossRef]

182. Wang, P.; Wang, F.; Lin, F.; Cao, Z. Identifying Peer-to-Peer Botnets Through Periodicity Behavior Analysis. In Proceedings of the 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications and 12th IEEE International Conference on Big Data Science and Engineering, Trustcom/bigdatase 2018, New York, NY, USA, 1–3 August 2018; p. 8455919. [CrossRef]
183. Barthakur, P.; Dahal, M.; Ghose, M.K. A framework for P2P botnet detection using SVM. In Proceedings of the 2012 International Conference on Cyber-enabled Distributed Computing and Knowledge Discovery, Cyberc 2012, Sanya, China, 10–12 October 2012; p. 6384967. [CrossRef]
184. Chen, Z.; Yu, X.; Zhang, C.; Zhang, J.; Lin, C.; Song, B.; Gao, J.; Hu, X.; Yang, W.; Yan, E. Fast botnet detection from streaming logs using online lanczos method. In Proceedings of the 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, USA, 11–14 December 2017; pp. 1408–1417. [CrossRef]
185. Ersson, J.; Moradian, E. Botnet Detection with Event-Driven Analysis. *Procedia Comput. Sci.* **2013**, *22*, 662–671. [CrossRef]
186. Almutairi, S.; Mahfoudh, S.; Almutairi, S.; Alowibdi, J.S. Hybrid Botnet Detection Based on Host and Network Analysis. *J. Comput. Netw. Commun.* **2020**, *2020*, 9024726. [CrossRef]
187. Lu, W.; Miller, M.; Xue, L. Detecting Command and Control Channel of Botnets in Cloud. In *Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments*; Traore, I., Woungang, I., Awad, A., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 55–62.
188. Zeng, Y.; Yan, G.; Eidenbenz, S.; Shin, K.G. Measuring the effectiveness of infrastructure-level detection of large-scale botnets. In Proceedings of the 2011 IEEE Nineteenth IEEE International Workshop on Quality of Service, San Jose, CA, USA, 6–7 June 2011; p. 5931312. [CrossRef]
189. François, J.; Wang, S.; Bronzi, W.; State, R.; Engel, T. BotCloud: Detecting botnets using MapReduce. In Proceedings of the 2011 IEEE International Workshop on Information Forensics and Security, Wifs 2011, Iguacu Falls, Brazil, 29 November–2 December 2011; p. 6123125. [CrossRef]
190. Saraubon, K.; Limthanmaphon, B. Fast Effective Botnet Spam Detection. In Proceedings of the 2009 Fourth International Conference on Computer Sciences and Convergence Information Technology, Seoul, Korea, 24–26 November 2009; pp. 1066–1070. [CrossRef]
191. Crespo, B.G.; Garwood, A. Fighting Botnets with Cyber-Security Analytics: Dealing with Heterogeneous Cyber-Security Information in New Generation SIEMs. In Proceedings of the 2014 Ninth International Conference on Availability, Reliability and Security, Fribourg, Switzerland, 8–12 September 2014; pp. 192–198. [CrossRef]
192. Derhab, A.; Bouras, A.; Muhaya, F.B.; Khan, M.K.; Xiang, Y. Spam Trapping System: Novel security framework to fight against spam botnets. In Proceedings of the 2014 21st International Conference on Telecommunications (ICT), Lisbon, Portugal, 4–7 May 2014; pp. 467–471. [CrossRef]
193. Tang, Y.; Cheng, G.; Yu, J.T.; Zhang, B. Catching modern botnets using active integrated evidential reasoning. *J. Internet Serv. Appl.* **2013**, *4*, 1–10. [CrossRef]
194. Yusof, M.; Saudi, M.M.; Ridzuan, F. A New Android Botnet Classification for GPS Exploitation Based on Permission and API Calls. *Lect. Notes Electr. Eng.* **2018**, *465*, 27–37. [CrossRef]
195. Mongkolluksamee, S.; Visoottiviseth, V.; Fukuda, K. Robust Peer to Peer Mobile Botnet Detection by Using Communication Patterns. In Proceedings of the AINTEC '18, AINTEC Asian Internet Engineering Conference, Bangkok, Thailand, 12–14 November 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 38–45. [CrossRef]
196. Karim, A.; Salleh, R.; Khan, K. SMARTbot: A Behavioral Analysis Framework Augmented with Machine Learning to Identify Mobile Botnet Applications. *PLoS ONE* **2016**, *11*, e0150077. [CrossRef] [PubMed]
197. Tidke, S.K.; Karde, P.; Thakare, V. Identification of Botnet hidden behind smartphone applications. In Proceedings of the 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, India, 1–2 August 2017; pp. 420–424. [CrossRef]
198. Garip, M.T.; Reiher, P.; Gerla, M. RIoT: A Rapid Exploit Delivery Mechanism against IoT Devices Using Vehicular Botnets. In Proceedings of the 2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall), Honolulu, HI, USA, 22–25 September 2019; pp. 1–6. [CrossRef]
199. Baltazar, J.; Costoya, J.; Flores, R. Steep rise in Koobface variants is boosted by social networking. *Comput. Fraud. Secur.* **2009**, *2009*, 19–20. [CrossRef]
200. Cybersecurity & Infrastructure Security Agency—"Alert (TA14-150A)—GameOver Zeus P2P Malware". Available online: https://us-cert.cisa.gov/ncas/alerts/TA14-150A (accessed on 4 August 2016).
201. Hatzivasilis, G.; Soultatos, O.; Chatziadam, P.; Fysarakis, K.; Askoxylakis, I.; Ioannidis, S.; Alaxandris, G.; Katos, V.; Spanoudakis, G. WARDOG: Awareness detection watchbog for Botnet infection on the host device. *IEEE Trans. Sustain. Comput.* **2019**, *6*, 4–18. [CrossRef]
202. Ahmed, Z.; Danish, S.M.; Qureshi, H.K.; Lestas, M. Protecting IoTs from mirai botnet attacks using blockchains. In Proceedings of the 2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), Limassol, Cyprus, 11–13 September 2019; p. 8858484. [CrossRef]
203. Yin, D.; Zhang, L.; Yang, K. A DDoS Attack Detection and Mitigation with Software-Defined Internet of Things Framework. *IEEE Access* **2018**, *6*, 24694–24705. [CrossRef]

204. Sadeghian, A.; Zamani, M. Detecting and preventing DDoS attacks in botnets by the help of self triggered black holes. In Proceedings of the 2014 Asia-Pacific Conference on Computer Aided System Engineering (APCASE), Bali, Indonesia, 10–12 February 2014; pp. 38–42. [CrossRef]
205. De Donno, M.; Donaire Felipe, J.M.; Dragoni, N. ANTIBIOTIC 2.0: A Fog-based Anti-Malware for Internet of Things. In Proceedings of the 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS PW), Stockholm, Sweden, 17–19 June 2019; pp. 11–20. [CrossRef]
206. Wang, P.; Wu, L.; Cunningham, R.; Zou, C.C. Honeypot Detection in Advanced Botnet Attacks. *Int. J. Inf. Comput. Secur.* **2010**, *4*, 30–51. [CrossRef]
207. Jerkins, J.A. Motivating a market or regulatory solution to IoT insecurity with the Mirai botnet code. In Proceedings of the 2017 IEEE 7th Annual Computing and Communication Workshop and Conference, CCWC 2017, Las Vegas, NV, USA, 9–11 January 2017; p. 7868464. [CrossRef]
208. Oliveri, A.; Lauria, F. Sagishi: An undercover software agent for infiltrating IoT botnets. *Netw. Secur.* **2019**, *2019*, 9–14. [CrossRef]
209. Tetarave, S.K.; Tripathy, S.; Kalaimannan, E.; John, C.; Srivastava, A. A Routing Table Poisoning Model for Peer-to-Peer (P2P) Botnets. *IEEE Access* **2019**, *7*, 67983–67995. [CrossRef]
210. Davis, C.R.; Fernandez, J.M.; Neville, S. Optimising sybil attacks against P2P-based botnets. In Proceedings of the 2009 4th International Conference on Malicious and Unwanted Software, Malware 2009, Montreal, QC, Canada, 13–14 October 2009; p. 5403016. [CrossRef]
211. Kelly, C.; Pitropakis, N.; McKeown, S.; Lambrinoudakis, C. Testing and Hardening IoT Devices against the Mirai Botnet. In Proceedings of the 2020 International Conference on Cyber Security and Protection of Digital Services (Cyber Security), Dublin, Ireland, 15–19 June 2020; p. 9138887. [CrossRef]
212. Hallman, R.; Bryan, J.; Palavicini, G.; Divita, J.; Romero-Mariona, J. IoDDoS—The internet of distributed denial of sevice attacks A case study of the mirai malware and IoT-Based botnets. In Proceedings of the IOTBDS 2017—2nd International Conference on Internet of Things, Big Data and Security, Porto, Portugal, 24–26 April 2017; pp. 47–58. [CrossRef]
213. Huan, L.; Yu, Y.; Lv, L.; Li, S.; Xia, C. A botnet-oriented collaborative defense scheme description language. In Proceedings of the 9th International Conference on Computational Intelligence and Security, CIS 2013, Emeishan, China, 14–15 December 2013; p. 6746511. [CrossRef]
214. Bradbury, D. Fighting botnets with sinkholes. *Netw. Secur.* **2012**, *2012*, 12–15. [CrossRef]
215. Kanich, C.; Kreibich, C.; Levchenko, K.; Enright, B.; Voelker, G.M.; Paxson, V.; Savage, S. Spamalytics: An empirical analysis of spam marketing conversion. *Commun. ACM* **2009**, *52*, 99–107. [CrossRef]
216. Watkins, L.; Kawka, C.; Corbett, C.; Robinson, W.H. Fighting banking botnets by exploiting inherent command and control vulnerabilities. In Proceedings of the 9th IEEE International Conference on Malicious and Unwanted Software, Malcon 2014, Fajardo, PR, USA, 28–30 October 2014; p. 6999411. [CrossRef]
217. Stone-Gross, B.; Cova, M.; Gilbert, B.; Kemmerer, R.; Kruegel, C.; Vigna, G. Analysis of a botnet takeover. *IEEE Secur. Priv.* **2011**, *9*, 5560627. [CrossRef]
218. The Spamhaus Project. Available online: https://www.spamhaus.org/bcl/ (accessed on 23 June 2020).
219. IBM X-Force exchange. Available online: https://exchange.xforce.ibmcloud.com/collection/Botnet-Command-and-Control-Servers-7ac6c4578facafa0de50b72e7bf8f8c4 (accessed on 23 June 2020).
220. Li, J.; Xue, Z. Distributed Threat Intelligence Sharing System: A New Sight of P2P Botnet Detection. In Proceedings of the 2nd International Conference on Computer Applications and Information Security, ICCAIS 2019, Riyadh, Saudi Arabia, 1–3 May 2019; p. 8769511. [CrossRef]
221. Dog, S.E.; Tweed, A.; Rouse, L.; Chu, B.; Qi, D.; Hu, Y.; Yang, J.; Al-Shaer, E. Strategic cyber threat intelligence sharing: A case study of IDS logs. In Proceedings of the 2016 25th International Conference on Computer Communications and Networks, ICCCN 2016, Waikoloa, HI, USA, 1–4 August 2016; p. 7568578. [CrossRef]
222. Eslahi, M.; Rostami, M.R.; Hashim, H.; Tahir, N.M.; Naseri, M.V. A data collection approach for Mobile Botnet analysis and detection. In Proceedings of the 2014 IEEE Symposium on Wireless Technology and Applications (ISWTA), Kota Kinabalu, Malaysia, 28 September–1 October 2014; pp. 199–204. [CrossRef]
223. Garcia, S.; Erquiaga, M.J.; Shirokova, A.; Garcia Garino, C. Geost Botnet. Operational Security Failures of a New Android Banking Threat. In Proceedings of the 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS PW), Stockholm, Sweden, 17–19 June 2019; pp. 406–409. [CrossRef]
224. Vidal-González, S.; García-Rodríguez, I.; Aláiz-Moretón, H.; Benavides-Cuéllar, C.; Benítez-Andrades, J.A.; García-Ordás, M.T.; Novais, P. Analyzing IoT-Based Botnet Malware Activity with Distributed Low Interaction Honeypots. In *Trends and Innovations in Information Systems and Technologies*; Rocha, Á., Adeli, H., Reis, L.P., Costanzo, S., Orovic, I., Moreira, F., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 329–338.
225. Safaei Pour, M.; Mangino, A.; Friday, K.; Rathbun, M.; Bou-Harb, E.; Iqbal, F.; Samtani, S.; Crichigno, J.; Ghani, N. On data-driven curation, learning, and analysis for inferring evolving internet-of-Things (IoT) botnets in the wild. *Comput. Secur.* **2020**, *91*, 101707. [CrossRef]
226. Borgaonkar, R. An analysis of the asprox botnet. In Proceedings of the 4th International Conference on Emerging Security Information, Systems and Technologies, Securware 2010, Venice, Italy, 18–25 July 2010; p. 5633693. [CrossRef]

227. Li, X.; Duan, H.; Liu, W.; Wu, J. The growing model of Botnets. In Proceedings of the 2010 International Conference on Green Circuits and Systems, Shanghai, China, 21–23 June 2010; p. 5543027. [CrossRef]
228. Patsakis, C.; Casino, F.; Katos, V. Encrypted and covert DNS queries for botnets: Challenges and countermeasures. *Comput. Secur.* **2020**, *88*, 101614. [CrossRef]
229. Stone-Gross, B.; Cova, M.; Cavallaro, L.; Gilbert, B.; Szydlowski, M.; Kemmerer, R.; Kruegel, C.; Vigna, G. Your Botnet is My Botnet: Analysis of a Botnet Takeover. In Proceedings of the 16th ACM Conference on Computer and Communications Security, CCS 09, Chicago, IL, USA, 9–13 November 2009; Association for Computing Machinery: New York, NY, USA, 2009; pp. 635–647. [CrossRef]
230. Barford, P.; Yegneswaran, V. An Inside Look at Botnets. In *Malware Detection*; Christodorescu, M., Jha, S., Maughan, D., Song, D., Wang, C., Eds.; Springer: Boston, MA, USA, 2007; pp. 171–191.
231. Shahrestani, A.; Feily, M.; Masood, M.; Muniandy, B. Visualization of invariant bot behavior for effective botnet traffic detection. In Proceedings of the 2012 International Symposium on Telecommunication Technologies, ISTT 2012, Kuala Lumpur, Malaysia, 26–28 November 2012; p. 6481606. [CrossRef]
232. Rowe, B.; Wood, D.; Reeves, D. How the public views strategies designed to reduce the threat of botnets. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer: Berlin/Heidelberg, Germany, 2010; Volume 6101, pp. 337–351. [CrossRef]
233. Khattak, S.; Ramay, N.R.; Khan, K.R.; Syed, A.A.; Khayam, S.A. A Taxonomy of Botnet Behavior, Detection, and Defense. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 898–924. [CrossRef]
234. Peterson, A. Yes, Terrorists Could Have Hacked Dick Cheneys Heart. Available online: https://www.washingtonpost.com/news/the-switch/wp/2013/10/21/yes-terrorists-could-have-hacked-dick-cheneys-heart/ (accessed on 14 December 2020).