electronics

MDPI

Article

A Survey of DDOS Attack Detection Techniques for IoT Systems Using BlockChain Technology

Zulfiqar Ali Khan *,† and Akbar Siami Namin †

Department of Computer Science, Texas Tech University, P.O. Box 43104, Lubbock, TX 79409-3104, USA
* Correspondence: zulfi.khan@ttu.edu
† These authors contributed equally to this work.

Abstract: The Internet of Things (IoT) is a network of sensors that helps collect data 24/7 without human intervention. However, the network may suffer from problems such as the low battery, heterogeneity, and connectivity issues due to the lack of standards. Even though these problems can cause several performance hiccups, security issues need immediate attention because hackers access vital personal and financial information and then misuse it. These security issues can allow hackers to hijack IoT devices and then use them to establish a Botnet to launch a Distributed Denial of Service (DDoS) attack. Blockchain technology can provide security to IoT devices by providing secure authentication using public keys. Similarly, Smart Contracts (SCs) can improve the performance of the IoT–blockchain network through automation. However, surveyed work shows that the blockchain and SCs do not provide foolproof security; sometimes, attackers defeat these security mechanisms and initiate DDoS attacks. Thus, developers and security software engineers must be aware of different techniques to detect DDoS attacks. In this survey paper, we highlight different techniques to detect DDoS attacks. The novelty of our work is to classify the DDoS detection techniques according to blockchain technology. As a result, researchers can enhance their systems by using blockchain-based support for detecting threats. In addition, we provide general information about the studied systems and their workings. However, we cannot neglect the recent surveys. To that end, we compare the state-of-the-art DDoS surveys based on their data collection techniques and the discussed DDoS attacks on the IoT subsystems. The study of different IoT subsystems tells us that DDoS attacks also impact other computing systems, such as SCs, networking devices, and power grids. Hence, our work briefly describes DDoS attacks and their impacts on the above subsystems and IoT. For instance, due to DDoS attacks, the targeted computing systems suffer delays which cause tremendous financial and utility losses to the subscribers. Hence, we discuss the impacts of DDoS attacks in the context of associated systems. Finally, we discuss Machine-Learning algorithms, performance metrics, and the underlying technology of IoT systems so that the readers can grasp the detection techniques and the attack vectors. Moreover, associated systems such as Software-Defined Networking (SDN) and Field-Programmable Gate Arrays (FPGA) are a source of good security enhancement for IoT Networks. Thus, we include a detailed discussion of future development encompassing all major IoT subsystems.

Keywords: vulnerabilities; blockchain; smart contracts; detection techniques; machine-learning; interplanetary file system (IPFS); IoT architecture; DDoS; denial of service

Citation: Khan, Z.A.; Namin, A.S. A Survey of DDOS Attack Detection Techniques for IoT Systems Using BlockChain Technology. *Electronics* **2022**, *11*, 3892. https://doi.org/10.3390/electronics11233892

Academic Editor: Jorge Bernal Bernabe

Received: 01 September 2022
Accepted: 07 November 2022
Published: 24 November 2022

1. Introduction

The demand for constant monitoring to make better decisions resulted in the innovation of Data Acquisition Systems [1]. Sensors are crucial to the success of Data Acquisition systems, which process the sensor data and collaborate with processors and communication devices to send the data to IoT gateways. The IoT network stores sensor data on clouds or the blockchain for more secure access. Otherwise, IoT devices have limited storage and processing capabilities and also suffer from connectivity problems [2]. These drawbacks make IoT devices easy victims of adversarial attacks. The most crucial of these attacks is

the DDoS attack because it hijacks many IoT devices by creating a BoTNet. The hijacked IoT devices may include several types: closed-circuit television cameras, routers, bulbs, digital video recorders, and so on [3]. These BoT-controlled devices help the attacker to achieve their subversive activities. The success of DDoS attacks lies in that thousands of compromised devices using varied IP addresses form a Botnet and then target a single computer, which ultimately chocks the network's bandwidth, preventing all valid requests from reaching the target computer. This network blockage causes DDoS problems because the said computer may be a server, and associated clients will not receive the required data, resulting in performance degradation.

Research work in [4] divides DDoS attacks into two stages: The first stage involves transforming the IoT devices into zombies by infecting them with malware. The second step involves controlling the infected devices with a command and control center. Finally, in the third step, the command and control center uses the zombies to launch a high-traffic attack on a targeted device. This flooding attack is the main characteristic of both DoS and DDoS attacks, but DDoS attacks use a command and control center to launch this attack [5].

Several technologies are being used to boost the internal capabilities of IoT devices, which are rapidly increasing in popularity. Table 1 presents an overview of the state-of-the-art contributions to various IoT subsystems.

The Key Contributions

(1) This is the first comprehensive examination of 13 IoT vulnerability detection systems (in detail, their general information, workings, and techniques), with 11 of them focusing primarily on DDoS attacks.
(2) In the light of the current emphasis on the blockchain, we categorize IoT-based vulnerability detection systems into blockchain-based categories.
(3) Through our discussion of DDoS attacks based on the bitcoin blockchain, Ethereum SCs, power systems, and UDP protocol, we extend our coverage beyond IoT-based DDoS attacks.
(4) We briefly outline the three-layer and five-layer architectures for IoT networks.
(5) We provide future directions for the advancement of research and the problems hindering the IoT and IoT subsystems in handling DDoS attacks.

Table 1. Contribution table: latest research pertaining to relevant sections.

Survey Section	Significance	The Focus of Research Papers
Motivation (Section 2)	Survey	– DDoS Mitigation Techniques [6] – Detection of DDoS Attack [7] – Blockchain-based Solutions [3]
Background Knowledge (Section 4)	Blockchain IPFS Machine Learning DDoS Power Grids	– Secure Blockchain Model for Botnet Detection [8] – IoT Data Streaming Using Blockchain and IPFS [9] – Data Security: IoT, Blockchain and IPFS [10] – DDoS Attack Prediction [11] – Modeling Impact of DDoS Attack [12]
Detection Techniques (Section 7)	Blockchain-based Collaborative Blockchain Non-Blockchain-based	– DDoS Detection for Blockchain Network Layer [13] – SDN Targeted DDoS [14] – Random Forest and Mutual Information-based DDoS Detection [15]

2. Motivation

DDoS attacks have become the latest modus operandi for acquiring illegal funds from established businesses and government agencies. Initially, hackers focus on financial institutions for stealing funds through online access. Credit card and debit card frauds are also common. However, the discovery of DDoS attacks has provided new avenues to hackers, as in the Colonial Pipeline ransomware attack discussed on Wikipedia. They do not have to perform direct intrusion into the victim's computer or hack their password to access

their bank account. Instead, hackers now receive technological assistance through botnets to succeed in their criminal plans. Through botnet-enabled attacks, hackers cripple the victim's computer's network by rocketing millions of requests to the victim's computer's IP address. This attack continues until the victim's computer comes to a halt. At that time, hackers phone the victim and ask him to pay the ransom in lieu of restoring his computer services. Therefore, our moral and legal duty is to curb such crimes or help stop them. One way to provide this service is to conduct research in the field of DDoS and provide technical information to the people so that they become astute enough to handle it. For this information dissemination, we are conducting this survey.

Comparison of Our Survey with State-of-the-Art Surveys

However, there are other surveys, as in [16–19], but the reasons which justify our research are as follows:

(a) The above survey-based endeavors have restricted scopes; for instance, the surveys in [16–18] focus on research papers related to Machine-Learning, whereas research in [19] focuses on the blockchain.
(b) On the other hand, our research focuses on all techniques related to DDoS attacks, including Machine-Learning and other methods such as security policies and traffic rates.
(c) Another distinguishing factor is that our research discusses DDoS attacks on the IoT and different IoT subsystems such as the blockchain, SCs, SDN, power grids, and networking protocols such as UDP.
(d) Table 2 compares state-of-the-art surveys, including ours, in the context of the focus of the surveys, the survey methodologies, and the DDoS attacks on IoT subsystems discussed in the survey paper. Thus, our research provides greater potential for learning and advancement of knowledge for both students and researchers.

Table 2. Salient Features of Surveys, Including Ours, where SDN means software-defined networking, BC means blockchain, PG means power grid, SC means smart contract, UDP means user datagram protocol.

Ref#	Focus of Survey: Methodology	Discussed DDoS Attacks on IoT Sub-System					
		SDN	BC	PG	SC	IoT	UDP
1. [3]	DDoS Mitigation Techniques for IoT using Blockchain: Strengths and weaknesses	✓				✓	✓
2. [16]	DDoS Detection Techniques for IoT using ML and deep learning: Attributes of detection technique					✓	
3. [17]	botnet Detection Approaches for IoT: Comparative study of botnets and technique, dataset, entity detected, devices used, etc					✓	✓
4. [18]	DDoS Detection Approaches using Deep Learning: Preprocessing details, experimental values, and setups	✓				✓	✓
5. [19]	DDoS Mitigation Technique using BC for IoT and SDN: Solution based on deployment location	✓				✓	✓
6. [20]	DDoS Detection, Mitigation, and Prevention Techniques for Networks based on protocols such as TCP, UDP, ICMP using SDN and programmable data planes (PDP): Attributes of detection techniques						✓
7. [21]	DDoS Detection Technique using ML for SDN: Traffic-Analysis-related experiments	✓				✓	✓
8. [22]	DDoS Detection Technique using ML for Network Services: Traceback-related experiments						✓
9. [23]	DDoS Detection Technique using ML for IoT: Description of ML techniques	✓				✓	✓
10. [24]	DDoS prevention using SDN and Blockchain for IoT and Network services: Important properties of DDoS attacks and defense techniques					✓	✓
11. Ours	DDoS Detection using Blockchain, SCs, ML: Description of General Information, processes and methods to tackle DDoS attacks		✓	✓	✓	✓	✓

3. Materials and Methodology

We continue our efforts to explore the security issues as we did in our previous research by pointing out vulnerabilities related to the IoT [25] and the Ethereum blockchain [26]. However, we have decided to focus on DDoS-based detection techniques due to several recent hacks involving DDoS attacks. At the same time, we also incorporate interesting research techniques for detecting physical and firmware attacks. Therefore, we divide our data collection endeavor into different steps, as discussed in Figure 1. The first step is to formalize the research title so we can avoid work repetition. Hence, we downloaded DDoS survey papers related to Hindawi and MDPI because of their free availability and the variety of titles. We use the search strings "DDoS Attack IoT mdpi" or "DDoS Attack IoT hindawi". To achieve our objective of a unique title, we compared our envisioned research paper's name with collected research papers and altered it accordingly so that it is unique and appropriate according to our research plan. Finally, we confirm our title using a Google search.

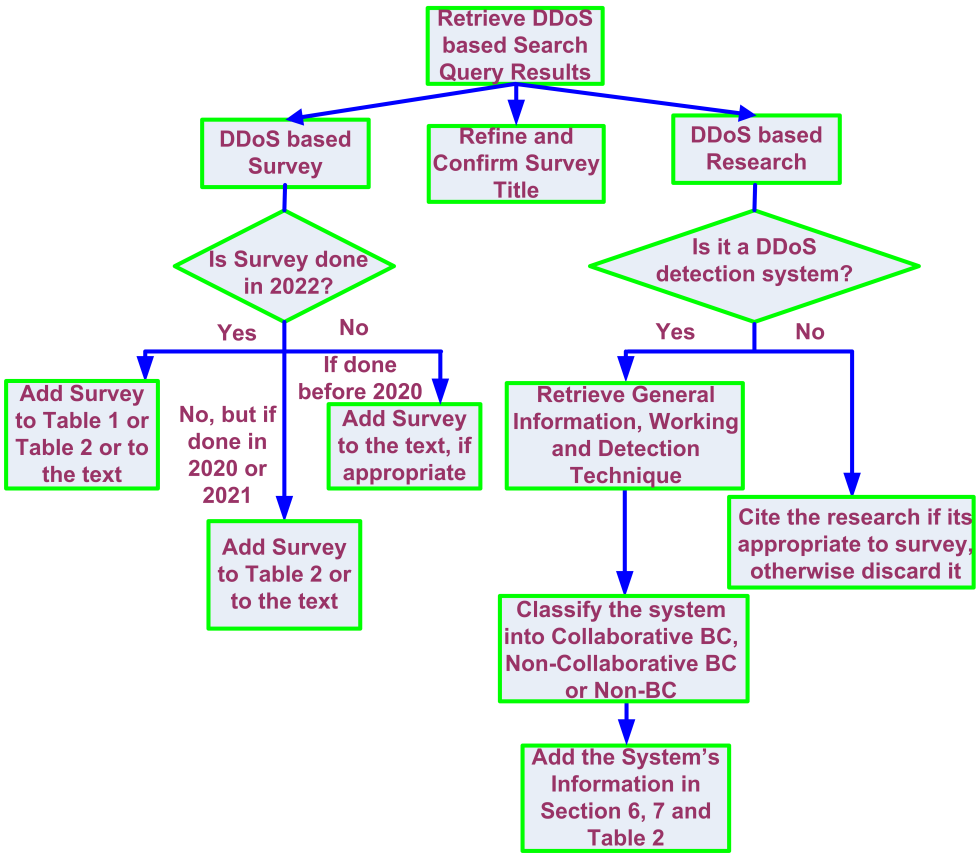

Figure 1. Flowchart for Data (i.e., Paper) Collection.

Our next objective is to focus on the novelty aspect of the research. We were aware of DDoS attacks on the blockchain, SCs, and TCP/IP protocols based on our previous study. However, we could not find any specific survey papers in connection with DDoS attacks on the above subsystems. Thus, our survey contains DDoS-attack-related information for blockchain, SCs, power grids, networking protocols, and IoT systems. We incorporate this essential knowledge about the scenarios of DDoS-based attacks in Section 4. Then, we downloaded research papers on firmware, physical, and DDoS-attack-based detection systems. However, we first focused on recent work using strings such as: "IoT Firmware Attacks 2022", "IoT DDoS Attacks 2022", and so on. We divided the downloaded papers based on the involvement of blockchain in those systems as discussed in the flowchart in Figure 1. We discuss DDoS systems in Sections 6 and 7. Surveyed literature on DDoS attacks helped us formulate Table 3 in which we compare DDoS attacks on IoT-subsystems, including Mirai. In addition, Table 4 summarizes the relevant DDoS detection techniques.

It is worth mentioning that we used our university's website to download ACM, IEEE, and Elsevier-related research papers. Finally, we collected data about different sections of our research paper and downloaded some Machine-Learning and IoT-based papers. We also downloaded some news related to DDoS-related ransomware attacks. The research work in [16,17] greatly helped refine our research methodology.

Exclusion and Inclusion Policy

Exclusion Policy: Our default exclusion policy is to avoid papers unrelated to IoT DDoS, IoT firmware, and IoT physical attacks.
Inclusion Policy: We allowed papers related to DDoS and the IoT, DDoS and blockchain, DDoS and SC, DDoS and power grids, and DDoS and UDP.

4. Theoretical Foundations or Background Knowledge

First, this section will briefly describe the blockchain and other systems, components, and networking protocols (UDP only) that have suffered DDoS attacks. Hence, we will also provide a brief description of these attacks, but not for Machine-Learning and SDN. However, we will summarize important Machine-Learning concepts, IPFS (Section 4.7), and IoT network architecture at the end of the discussion of the DDoS attacks.

4.1. Blockchain

The blockchain is a peer-to-peer network of mining nodes. Each node maintains its copy of monetary transactions as a ledger and validates the ongoing transactions using the consensus mechanism incorporated into the blockchain. After validation, the transaction is subject to a cryptographic hash algorithm, the transaction's hash becomes part of the ledger, and the process continues. The hash is essential in linking the blocks like a chain, so each block contains its hash and the next block's hash. The advantage of this decentralization is shareable accountability, which makes it harder for the attacker to tamper with the ledger. The attacker requires the approval of the majority of the nodes, which could be in the thousands, to alter the immutability of the ledger.

4.1.1. Advantages of Blockchain for IoT Networks

The following are the advantages of the blockchain-based approach [27]:

(1) The blockchain performs device authentication. Thus, the blockchain prevents illegal access to IoT data.
(2) The blockchain records transactions and SCs can perform processing; hence, the merger of the two can perform both storage and processing at no additional cost.
(3) The blockchain has several nodes for validation purposes called miners. Thus, tracking anomalies such as the unprecedented volume of data flow from IoT devices and timing is more comprehensive than other platforms.

4.1.2. DDoS Attacks in the Bitcoin Blockchain

According to one news source, discussed in [28], a blockchain platform recently became the epicenter of the most historically significant DDoS attacks. This situation is alarming for both Bitcoin and Ethereum. Thus, to avoid the repetition of such attacks, it is necessary to understand the exploits of the related platforms to prevent future episodes. Let us see how the attackers generated DDoS attacks on the Bitcoin platforms.

The Bitcoin blockchain, like other blockchains, does not need a trusted third party for transactions between its nodes. Similarly, Bitcoin allows secure and tamper-proof transactions because one can verify them. However, its public nodes, which connect to the Internet, threaten to become a victim of DDoS attacks. Two types of DDoS attacks are possible: the UDP attack, discussed in Section 4.4.1, and the transaction flood attack, which we will discuss here.

In the transaction flood attack, the attacker sends several spam transactions with proper relay fees and mining fees, prioritizing the transactions for a block. However, this

fills the block, preventing real users' actual transactions from going into the block. Instead, they go into the mempool. The mempool stores the unconfirmed transactions until the block finishes, and Bitcoin allocates a new block for the chain. For example, suppose Bitcoin verifies 3–7 transactions in one second. Thus, if the mempool receives less than two transactions/second, there is no queue of unconfirmed transactions. However, once the mempool receives more than seven transactions/sec, the mempool develops a queue. There is a record of mempool's most extensive queue, which was created on 11 November 2017 [29]. It had USD 700 million worth of transactions, and the users paid higher mining fees to prioritize their transactions.

4.2. Smart Contracts (SCs)

Smart Contracts (SCs) further enhance blockchain technology. SCs transform traditional contracts or agreements (such as rental agreements) into a computer program deployed on the blockchain. Hence, SCs execute the contractual promises on the blockchain automatically. In addition, this digital activity reduces the overall business cost because intermediaries (such as real estate agents) have no involvement.

4.2.1. DDoS Attacks on SCs

SCs contain specific methods, and the execution of these methods results in a transaction with a gas cost paid upfront. Hence, a DDoS attack on an SC can occur by creating a high volume of transactions which slows down the Ethereum virtual machine (EVM), but this would be costly in terms of gas cost [30]. Conversely, if the attacker delivers an invalid transaction, in other words, does not pay the full fee, the EVM will reverse it [31]. However, the links in [32,33] discuss the Account Bloat Attack, in which the attacker sends a large volume of transactions by creating empty accounts on the Ethereum blockchain, which slows down the processing. The empty accounts mean that the attacker deploys the SC with zero Ether. This situation relates to when Ethereum charges a meager gas fee.

4.3. DDoS Attacks on Power Grid Systems

Power systems incorporate networking and digital technologies, transforming them into network or smart grid systems. This innovation has blessed the power system with advanced information processing and sensing techniques to control [34], measure, and distribute the grid outputs [35]. However, the dark side of these networking advancements is that the communication channels face the problem of false data injection caused [36] by the hijacking of sensors and the interruption of communication devices through network-based DDoS attacks. Thus, the intelligent grid operations suffer, which can even result in area blackouts. Therefore, the detection of DDoS attacks becomes the need of the hour to invoke a defensive system so that we can run our power and smart grid systems continuously and without disruptions. Work in [37] focuses on three indicators of compromise responsible for the DDoS attacks: (1) The response size of the power request, which would become abnormally large, hence signaling that a DDoS attack is in progress; (2) a mismatch in the application's port number, which means the use of an unusual port [38] instead of a well-known port (e.g., port#80) by an application server such as HTTP [38], thus being an indication of a DDoS attack before assuming that the unusual port number is under the influence of an attacker's bot, we must first confirm that this port is sending enormous traffic to the targeted server; and (3) locating an IP address sending multiple packets to the targeted server in a short time interval is also a sign of a DDoS attack.

A power plant may face delays in receiving control signals, which may also arise due to a DoS attack. Work in [39] states that attackers can maliciously induct random delays in control signals by jamming the network communication, and such delay attacks quickly spread out to the connected networks. Yan [40] studied this problem using an H_∞-based mathematical model to synthesize the controllers. The resulting distributed delay system differentiates between standard and delayed signals. It then obtains new controller conditions by applying the Lyapunov method, which can guarantee the stability of the power plant for a given performance. Earlier work in [41] studied the application of

the Lyapunov function in the context of a Non-fragile controller, which does not suffer from the implementation-based inaccuracies [42]. Again, the authors [41] use an integral-based Event Triggered System that reduces transmission errors, but integral-based approaches cause approximation errors. Authors applied the Bessel–Legendre inequality to handle approximation errors. However, the work in [43] also applies the Bessel–Legendre inequality, and work in [40] reports approximation errors in connection with the Bessel–Legendre inequality.

4.4. UDP

UDP stands for user datagram protocol. It has some low-overhead features. For instance, UDP is a connectionless and unreliable protocol, but these features make UDP ideal for chatting and VoIP [44]. Connectionless means there is no need to establish a session before sending data; hence, message loss is possible. On the other hand, unreliability means that UDP does not send an acknowledgment for received messages.

4.4.1. DDoS Attack Due to UDP

When a UDP [45,46] server receives the message, the server retrieves the message's port number and transfers the message to the process identified by the port number. However, if there is no process for receiving packets on that port number, the server replies with an ICMP error message indicating that the destination is unreachable. The attacker takes advantage of this routine and thus sends an enormous flood of UDP packets to the server, which requires ICMP responses. The generation of the ICMP responses to each received UDP packet quickly exhausts the server's resources, resulting in a DDoS attack because the attacker uses a botnet to generate messages.

4.5. Comparison of DDoS Attacks on IoT Sub-Systems and Mirai, Section (4.6.1)

Table 3 compares the DDoS attack on Bitcoin (Section 4.1.2), DDoS attack on Ethereum SCs (Section 4.2.1), DDoS attack on power grid systems (Section 4.3), DDoS attack due to UDP (Section 4.4.1), and the DDoS attack carried out by the Mirai botnet (Section 4.6.1). We cannot find any built-in power grids or UDP mechanisms to stop DDoS attacks. However, a system reboot can help to eliminate the attack. Figure 2 also highlights the impact of DDoS attacks on different IoT subsystems.

Table 3. Comparison of DDoS Attacks on IoT Subsystems and the Mirai Attack.

Attack Name	Impact	Recent Instance	Built-In Stopping Mechanism
SC DDoS	Exceeded SC's gas ceiling	2016 [32]	gas fee
Bitcoin blockchain DDoS	Flooding of transactions	2017 [29]	transaction fee
UDP DDoS	Exhausted resources of targeted server	2022 [47]	nill
Power Grids DDoS	Lack of power to huge population	2015 [48]	nill
Mirai	Shutdown of several important websites	2016 [17]	n/a

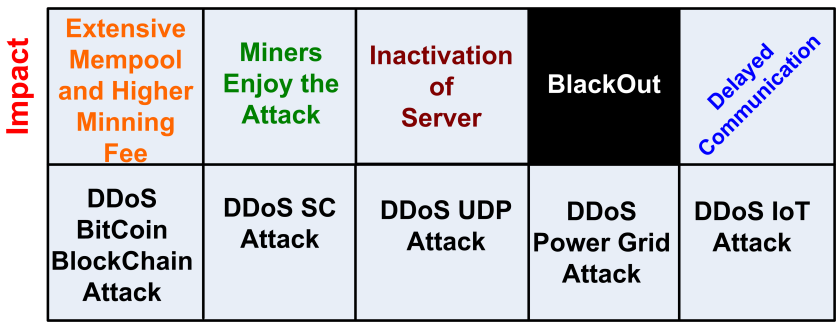

Figure 2. After Effects of DDoS Attacks.

4.6. Recent Trends in DDoS Attacks

A DDoS attack is a peculiar attack that blocks network communication by bombing it with abnormally high requests; botnet networks shelled 15.3 million requests in the latest attack discussed in [28]. Recently, DDoS attacks have become a major source of business for criminals. These criminals use DDoS attacks to jeopardize a system and then ask the victims to pay them a ransom fee to restore their system. However, the articles in connection with Ransom DDoS attacks [49], and specifically [50], state that the attackers may demand payment before launching the attack. Using cryptocurrency makes it impossible for law enforcement agencies to track transactions.

4.6.1. Mirai DDoS Attack

Mirai is an IoT malware that results in a hazardous DDoS attack. Mirai starts its crusade by infecting home routers, DVRs, and CCTV cameras by hacking their factory-installed credentials. Mirai creates a force of half a million IoT devices [51] to unveil a successful DDoS attack, which results in the crashing of several important websites. Mirai consists of the following four components [52]:

(i) A command and control module that allows a human to control the bots;
(ii) The Mirai bot runs on infected IoT devices and consists of three modules: (a) a scanner, which scans new vulnerable IoT devices and informs the reporting server; (b) the killer, which kills other malware competing with Mirai; and (c) the attacker, allowing the bot to hack other IoT devices when the command and control center orders it;
(iii) The reporting server interacts with the Mirai botnet to obtain information about the vulnerable IoT devices and passes it to the load server.
(iv) The loader server replaces vulnerable IoT devices' codes with the Mirai malware's codes, thus inducting vulnerable IoT devices into the botnet.

4.7. IPFS

IPFS is a decentralization technique similar to blockchain, but in this case, the file size is not restricted by the length of a block as in the EVM. The system divides larger files into smaller blocks of 256KB. The system also creates a reference object that hooks up all the blocks containing the file's data in a linked manner [53]. IPFS also integrates the concept of hashing. Thus, the system assigns each file in IPFS a hash value, which is the permanent address of the file.

4.8. Software-Defined Networking (SDN)

SDN uses programs and APIs to control networking devices and other hardware components [54]. Thus, we can have a virtual network to create traditional routers and switches, and we can change the configurations and network capacity using a centralized server. SDN categorizes [55] the handling of packets into three planes, which are discussed below [56]:

(i) The data plane deals with data packets and performs actions on them. Thus, the data plane works with line-speed [57] and interacts with the control plane through tables to obtain the required information.
(ii) The control plane provides the required information to the data plane so that the data plane can process and forward the data packets. For this purpose, the control plane creates tables such as the IP routing table and then adds, removes, and changes the table's entries, representing the routes to network destinations.
(iii) The management plane is responsible for configuring and monitoring network devices such as switches and routers.

4.9. Field-Programmable Gate Array (FPGA)

An FPGA is a programmable device. It allows the creation of a specific hardware circuit by programmatically combining several configurable blocks from among the thousands of identical available blocks within an FPGA system. The IoT is an evolving network of sensors, switches, routers, and other electrical and electronic devices and networks. As

a result, an FPGA can help in the IoT's development by reducing power consumption or adding networking features [58]. For instance, an FPGA can support the implementation of Transmission Control Protocol/Internet Protocol (TCP/IP), control, and data acquisition systems in the IoT [59]. However, the real benefit of an FPGA is that it can provide up-to-date hardware security to IoT devices through reprogrammable features, as well as its ability to simulate other hardware components [60].

4.10. Machine-Learning Performance Metrics

The final part of Machine-Learning research is to evaluate the performance of the Machine-Learning algorithms on the applied dataset. One can divide the issues related to performance metrics into classification and regression. However, based on the surveyed literature, we are only focusing on the metrics associated with classification problems. The commonly used terms in this regard are described below.

4.10.1. Precision

Precision is the number of correct or true positive results out of all the assumed positive results predicted by the classifier, i.e., both true and false positives. For instance, if a blindfolded Courser shot at its prey $6 + 10$ times, this means that the bullet hit the target six times (true positive), but the Courser missed ten times but assumed that the target was hit (false positives) [61]. Hence, the precision is $6/16 \approx 38\%$.

4.10.2. Recall

Recall is the number of true positives out of all actual positives, consisting of all samples that were detected and that should have been detected as positives. For instance, a blindfolded Courser hit the target $6 + 2$ times: the Courser heard only six of those hits (true positive) while the he did not hear the other two at all and therefore thought he did not hit them (false negative) [61]. Therefore, listening to a bullet hit has a recall of $6/8 = 75\%$.

4.10.3. F1 Score

Precision highlights how many false positives are in the result, and recall tells us how many false negatives are in the result. Finally, the F1 score is a method of combining false positives and false negatives using the harmonic mean [62].

4.11. Machine-Learning Algorithms

4.11.1. Random Forest

Random Forest helps in classification and regression problems. Classification helps to identify to which group our observation belongs. On the other hand, regression computes the relationship between the dependent and independent variables. For example, a regression can help determine the relationship between a man's food intake and his weight.

4.11.2. XGBoost

XGBoost stands for extreme gradient boosting. The purpose of gradient enabling is to improve a weak model by combining it with other weak models, and the process ultimately generates a robust model [63].

4.11.3. K-Nearest Neighbor

K-Nearest Neighbor (KNN) is a supervised learning algorithm. KNN uses labeled data for training but makes an appropriate prediction about the classification of unlabeled data during the testing phase. Here, 'K' represents the number of data groups a data point may belong to during the test [64].

5. A Glimpse of IoT Architecture

We have provided information about the Mirai botnet attack, which crippled several networks such as Twitter, GitHub, and Airbnb [65] by taking control of various IoT devices

such as IP cameras and routers (old version). Thus, IoT devices can serve as an entry point [66] to the attacking botnets. Hence, before discussing the DDoS detection techniques, it is a good idea to become familiar with the IoT network architecture, a collection of networking protocols and devices that may contain subsystems such as sensors, actuators, cloud storage devices, and blockchain storage devices. This section gives a brief description of IoT architectures.

There is no standard IoT architecture [67] or stack of protocols to govern the IoT's components. However, it is feasible to alter the TCP/IP protocol stack and thus define the layers of an IoT architecture [68]. Unlike TCP/IP, which deal with wired networks, IoT-based systems deal with wireless networks. Based upon this disparity, there are two popular architectures: three- and five-layer architectures, as shown in Figure 3 below.

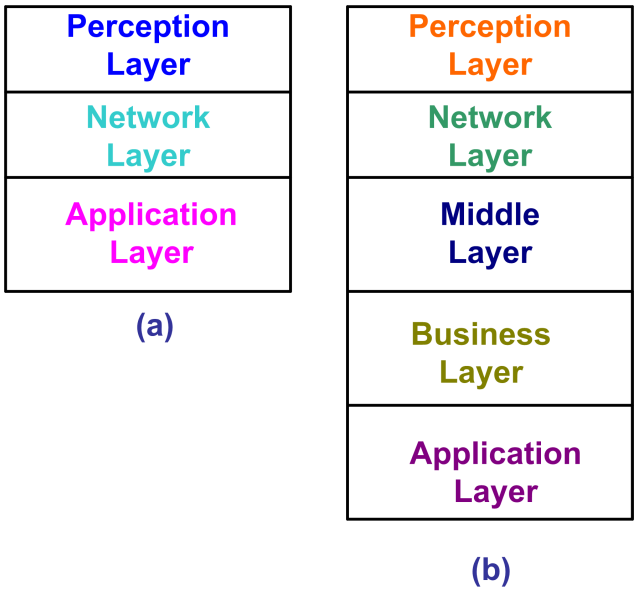

Figure 3. (**a**) Three-Layer Architecture; (**b**) Five-Layer Architecture.

5.1. Three-Layer Architecture

This architecture [67,69] is the simplest and the oldest architecture that can be adapted for IoT platforms. It consists of three layers:

5.1.1. Perception Layer

This physical layer interacts with its surroundings by collecting data through sensors. In recent advancements, IoT developers have endowed sensors with processing and intelligence capabilities and subsequently referred to them as smart objects or Smart "*Things*". Smart Things communicate with each other, requiring them to have IP addresses. Smart Things have three additional properties: (1) collecting the data related to their surroundings, (2) converting the data into electrical signals, and (3) identifying the surrounding objects without contact. As stated earlier, Smart Things need sensing, actuation, and radio frequency (i.e., RFID) or camera-based identification subsystems to achieve the above properties.

The IoT enables communication through wireless sensor networks (WSN) along with RFID. A typical RFID consists of: (1) tags that store identification information, and (2) readers which perform read and write operations on tags. A combination of tags, readers, and a background database helps RFID systems monitor the surrounding objects. On the other hand, WSNs consist of sensor nodes connected with a sink that interfaces with the Internet. Thus, sensors capture the surrounding data and help in surveillance and security applications [70]. Cameras are also a good source of object detection [71] by employing image processing techniques. For example, refrigerators embedded with

cameras [72] allow consumers to look inside the fridge while shopping in a grocery store to replenish items.

5.1.2. Network Layer

This layer interfaces the data received from the perception layer to the application layer. The network layer solves the addressing issues using the 6LoWPAN protocol [73], a variant of IPv6 for wireless sensor networks.

5.1.3. Application Layer

This layer analyzes and processes the data gathered from previous layers and then forwards it to IoT devices such as wearables, stopwatches, or smart fridges. It manages vast amounts of data using cloud computing through a service-oriented architecture (SoA). The SoA architecture allows us to pay only for the services we use. In addition to SoA, the application layer enables edge-computing technologies that provide resources near the data source instead of data centers.

5.2. Five-Layer Architecture

In this type of architecture, there are five layers. The first three layers, i.e., the perception layer, network layer, and application layer, are similar to the three-layer architecture and are discussed in Section 5.1. This section will discuss only the two additional layers, i.e., the business layer and the middle layer.

5.2.1. Business Layer

This layer caters to the needs of the business logic of the application. Business logic focuses on the E-commerce enterprise's rules related to data creation, alteration, storage, and presentation to the people [74]. In short, it informs the developer how it handles the data [75].

5.2.2. Middle Layer

This layer processes the data received from the previous layers and plays a vital role in standardization. It uses standard interfaces and protocols to solve the compatibility problem among different technologies (e.g., AI, Machine-Learning, RFID, etc.), as well as infrastructures designed for the healthcare, industrial, and retail sectors.

6. Brief Description of IoT Vulnerability Detection Systems

This section provides a short description of systems aiding firmware, physical, and DDoS attack detection. The reason for incorporating systems that support DDoS attack detection is to help future research advancements.

6.1. IoTCop

6.1.1. General Information

IoTCop [76] focuses on permissioned, blockchain-like hyperledger networks to monitor the proposed security framework and the devices within the framework. IoTCop protects the framework from security breaches such as the runtime update of security policies, which does not occur unless the hackers infect the majority of peers. IoTCop uses blockchain's consensus protocol to enforce the majority rule. Similarly, IoTCop monitors the network messages of IoT devices to check if the messages comply with security policies. For instance, one security policy requirement is that the messages should have a unique ID. IoTCop blocks the outbound traffic from a compromised device, i.e., an IoT device not adhering to security policies. However, IoTCop also provides 'no security' or 'minimum security' variants. In the case of the no security option, IoTCop does not enforce any security policy for the device's messages, but in the case of the minimum-security policy, IoTCop requires digitally signed messages. IoTCop does not support message encryption and also suffers from scalability problems. IoTCop detects firmware and physical attacks.

6.1.2. Working

The first step is to install the IoTCop client on the IoT device. If the device does not support IoTCop, then one can pair the device with an add-on hardware module supporting IoTCop. Once the device sends a message, the message is received by the sender interface of the IoTCop. The sender interface sends the message to all the IoTCop clients associated with the peer devices in the IoTCop framework. Each device determines whether or not the message is compliant with security policies. The result is sent to the destination device's receiver interface, and the destination device accepts the message only if the message is compliant with security policies.

6.2. Lightweight Collaborative Blockchain-Based Model

6.2.1. General Information

This work [77] discusses a blockchain-based anomaly detection model to perform distributed and collaborative anomaly detection on IoT devices. The authors used an extensible markup model (EMM) for anomaly detection by monitoring the application's jump sequence, which helps create a pattern. The jump sequences are related to accessing different memory locations when the application executes the functions, loops, and `if` statements. This monitoring helps to understand the application's normal flow and detects anomalies if the flow changes. The authors also determine a decentralized collaborative training method. They evaluated the performance of their model by monitoring smart camera applications.

6.2.2. Working

In the first phase, IoT devices collaborate to train an anomaly detection model on an IoT device by incrementally exchanging unique benign behaviors. Then, the devices continue their collaboration using blockchain's consensus mechanism, enabling protection and continuous updates in the model (with newer benign behaviors) and ensuring that IoT devices can validate access to the model.

6.3. IoT Agent and SC-Based DDoS Detection System

6.3.1. General Information

This work [78] focuses on SCs to detect DDoS attacks. SCs collaborate with agents; the agents are the nodes of a private network installed on the network's gateway. Agents oversee the network traffic and communicate with each other by exchanging path messages. Path messages store the total outgoing packet rate per destination IP address. Each agent concatenates the path messages with local information about the outgoing packet rate related to the monitored IoT devices. If the path messages reach the maximum size, then the agents transfer the path messages to the SC, which performs the DDoS attack detection by inspecting the local path messages. SC also triggers the generation of new path messages, validates the path messages, and blacklists the malicious agents.

6.3.2. Working

The first step is to install an agent on each IoT installation. Agents communicate path messages with each other by randomly choosing a destination agent. The receiver agent adds its installation's path message to the received message, and transfers the updated path message to another agent. When the path message reaches its critical length and the agent notices this event finds attack information in the path message, the agent sends an alert to all agents. Finally, the agents implement a consensus mechanism to detect the victim in a global state.

6.4. A Hybrid Deep-Learning-Based Mechanism for a Smart Transport System

6.4.1. General Information

This research [79] uses a secure blockchain for protecting transport data against DDoS attacks. SCs store transactions in the blockchain related to the local transport system and

its maintenance and supplier teams. The model uses a deep-learning-based AI system for DDoS attack detection, fusing an autoencoder with a multi-layer perceptron. The trained autoencoder performs the feature extraction, whereas the multi-layer perceptron categorizes DDoS attacks into different categories using softmax as an activation function in the final layer. The autoencoder uses unsupervised learning and helps to identify different types of data representation. The multilayer perceptron is a feed-forward network, i.e., the data flow direction is from input to output. The multi-layer perceptron consists of one input, one output, and multiple hidden layers, whereas the autoencoder consists of one input, one output, one bottleneck, and two hidden layers. The computing power of the multilayer perceptron depends upon the hidden layers. On the other hand, the bottleneck layer consists of fewer nodes than the previous layer and helps reduce dimensionality. All hidden layers use RELU activation functions. In addition, the dataset is subject to preprocessing to remove blank, irrelevant, or garbage values. The research achieves dataset balancing using Random Undersampling or/and Synthetic Minority Oversampling Techniques. Random undersampling removes imbalances in the dataset by randomly drawing the majority class's instances [80]. In Synthetic Minority Oversampling, minority class data are generated either by duplication or by developing new examples from the data. The detection involves decoding, which reconstructs the output and then compares it with the input.

6.4.2. Working

The authors deploy a SC to perform transactions related to the transport maintenance system. However, for DDoS detection, they first use an autoencoder for feature extraction from the datasets (i.e., CICDDoS2019, CICIDS2017, and BoT-IoT). Once the autoencoder shows a suitable data representation by reconstructing an output that resembles an input signal, authors use the autoencoder to initialize the multilayer perceptron. We believe this is their work's known drawback, as discussed in [81].

6.5. DDoS Detection by XGBoost, and Random Forest, in an SC-Based Blockchain–IoT System

6.5.1. General Information

This work [82] proposes a secure, SC-based blockchain interacting with IoT clusters. The system employs a distributed fog computing network using an intrusion detection system (IDS), i.e., the IDS is part of the fog computing network to safeguard the confidentiality of IoT data and mask DDoS attacks on SCs. Furthermore, the distributed fog computing network interconnects the blockchain and the IoT system. Thus, fog computing performs an additional network security task compared to its typical deployment for communication, data storage, and processing by keeping the network closer to the source. The fog computing system has two additional features: (1) the interplanetary file system (IPFS) (Section 4.7), which helps in load balancing, and (2) preprocessing the data and applying AI techniques such as Random Forest (Section 4.11.1) and XGBoost (Section 4.11.2), on the IoT data to predict the attack vectors and strengthen the intrusion detection system.

6.5.2. Working

Sensors collect the data and send it to the interplanetary file system (IPFS) (Section 4.7), which operates the same way as a blockchain but uses the cloud and other publicly available storage spaces to store the data [83]. Next, a combined intrusion detection system and SC (IDS-SC) evaluates the IPFS (Section 4.7) data that utilize AI-based approaches such as XGboost (Section 4.11.2) and Random forest (Section 4.11.1).

6.6. Distributed Intrusion Detection System (IDS) to Detect DDoS Attacks in Blockchain–IoT Network

6.6.1. General Information

The authors [84] discuss the detection of DDoS attacks on memory pools in connection with the Bitcoin blockchain. The authors used the same approach discussed in their previous work [82]. Both techniques use fog computing, IDS, XGBoost (Section 4.11.2), and

Random Forest (Section 4.11.1). The authors do not compare this Bitcoin-related DDoS detection technique with their previous approach for detecting DDoS attacks on SCs. Both the SCs and Bitcoin technologies create pools of transaction requests. The pools contain both low-priced and high-priced transactions. Miners process the high-priced transactions first, leaving the low-priced transactions to clot in the pool. The attacker takes advantage of this prioritization and overloads the pool with more low-priced transactions, resulting in Bitcoin's transaction processing coming to a standstill. This is the account bloat attack as discussed in [32].

6.6.2. Working

First, the system creates a Machine-Learning-based intrusion detection system by training the dataset using the Random Forest (section 4.11.1) and XGBoost (Section 4.11.2) algorithms. The trained intrusion detection system then processes the data collected by the IoT clusters. The trained system detects if the transaction is normal or abnormal. The detection system forwards normal transactions to the miners for validation and then to the blockchain to add them to the distributed ledger. The abnormal transactions go to the administrator for further processing.

6.7. Grammar-Based Filtering and Clustering Algorithm for DDoS Detection

6.7.1. General Information

This research [4] utilizes a hybrid security strategy to confront DDoS attacks. The authors filtered the packets using a deep packet inspection and clustering algorithm using grammar-based and Machine-Learning algorithms. However, the authors did not provide more details about packet inspection and the Machine-Learning model. This research focuses on TCP flood attacks. For the evaluation of system performance metrics, the authors investigate precision (Section 4.10.1), sensitivity, and F1-score (Section 4.10.3). For measuring security performance, the authors calculate the total average time in the system during an attack using queueing theory, which states that the mean wait in the system is the sum of the mean wait in the queue and the service time [85].

6.7.2. Working

Authors first define the security policies to confront the DDoS attack. The policy focuses on preventing TCP-flood attacks, malware propagation, and filtering non-MQTT and illegitimate telnet packets. In the first phase, deep packet investigation filters out the packets not associated with TCP and MQTT services using viral DB [86] and packet sniffing techniques. Packet sniffing focuses on retrieving the header information. On the other hand, viral DB contains information about malware signatures. In the second phase, parsed packets are subject to Machine-Learning algorithms to reject suspicious packets related to the DDoS attack.

6.8. DDoS Detection Using Machine-Learning and SMOTE-Based Techniques (SMOTE)

6.8.1. General Information

This research [87] focuses on detecting DDoS attacks on IoT networks using Machine-Learning techniques such as K-Nearest Neighbor (KNN) (Section 4.11.3), a Naive Bayes model, and a multi-layer perception artificial neural network. The authors used the Bot-IoT dataset for training purposes and measured the performance of the algorithms using accuracy, precision (Section 4.10.2), recall (Section 4.10.1), F1-score (Section 4.10.3), and ROC AUC metrics. During data preprocessing, the authors perform: (1) data cleansing for fixing and removing incomplete information; (2) normalization for transforming the data into a scale of [0, 1]; and (3) convert the non-numeric data, such as protocol names, into numeric values, e.g., value '1' replaces TCP. For feature engineering, the authors use chi-square to select the appropriate features. The synthetic minority over-sampling technique helps create a balanced dataset by choosing a data sample and finding its nearest neighbors. The system multiplies the resultant value by generating random numbers between 0 and

1 to obtain more sample data. The authors also apply cross-validation techniques for performance evaluation in which the authors split the dataset into five subsets. Then, the authors use a subset as a test set and the remaining subsets as a training set. This technique helps the authors better evaluate the performance of their Machine-Learning algorithms.

6.8.2. Working

First, the authors analyzed the botnet behavior by retrieving packet information. Then, the authors used manual and top F-score techniques (also known as chi-square) for feature selection and extraction. The method requires calculating the chi-square for all features and the mean value of all features. Finally, the authors selected the features more significant than the mean value. Then, the authors removed the imbalanced dataset using the SMOTE technique and trained their KNN model (Section 4.11.3) using the BoT-IoT dataset.

6.9. Machine-Learning-Based Smart Detection System

6.9.1. General Information

This smart detection system [88] works on the identification of DDOS attacks using Machine-Learning techniques such as XGBoost (Section 4.11.2), AdaBoost, Random Forests (Section 4.11.1), and multilayer perceptron. They focused on fewer features and used less memory to reduce complexity and improve the system's efficiency. Their performance metrics consist of accuracy, recall (Section 4.10.2), precision (Section 4.10.1), and F1 score (Section 4.10.3). The authors used the CICIDS2017 dataset to train the network.

6.9.2. Working

Working starts by building a classification model which can classify the DDoS attack attributes as benign or malicious after training on the CICIDS2017 dataset. The authors used the recursive feature elimination algorithm to reduce the dimensionality of the data [89]. The algorithm helps to remove the replicated feature and makes the data less detailed. For automation, one can use Python's sklearn tool [90]. The algorithm calculates the importance of features in the dataset and removes the less critical features in each iteration until the number of features reaches a threshold value. The next step employs the classification model to identify the malicious DDoS attacks.

6.10. IoT-Based Monitoring System for Banking Sector

6.10.1. General Information

The work in [91] used an open-source platform's training data related to DDoS attacks. After preprocessing and feature extraction, the authors split the data into training and testing datasets. For training purposes, the authors trained the dataset on three Machine-Learning algorithms, i.e., K-Nearest Neighbor (KNN) [92] (Section 4.11.3), Support Vector Machines (SVM), and Random Forests (Section 4.11.1). The NN consists of the input layer retrieving the input signal, the hidden layers representing the neighbors, and the output layer representing the prediction. Finally, the authors calculate the system's performance using accuracy, precision (Section 4.10.1), recall (Section 4.10.2), and F1 score (Section 4.10.3).

6.10.2. Working

The authors preprocessed the dataset to remove null values and to scale and balance the dataset. Next, the authors applied feature engineering to collect the best features from the data and split it into training (70%) and testing (30%). Finally, the authors trained the modified models of SVM, KNN (Section 4.11.3), and Random Forest (Section 4.11.1).

6.11. A Deep Blockchain Framework-Enabled Collaborative Intrusion Detection System

6.11.1. General Information

The authors [93] developed a collaborative intrusion detection system based on a deep blockchain framework (DBF). The system has four main parts: (1) First is a cloud vendor, which is the same as a cloud data center, (2) Second is a privacy-preservation-based

blockchain and SC. The blockchain helps in two ways: first, it provides SCs which specify the policy for sending and using intrusion detection system alerts. Despite these SC rules, the authors also provide a trusted execution environment (TEE) to protect SC from unauthorized users because the SCs' bytecodes are visible to all in public blockchains. At the same time, the blockchain also stores the warning messages generated by the detection system and the entire cloud transactions using the consensus protocol, thus ensuring the validity of cloud transactions. Furthermore, the warning messages help train IDS. For learning, the authors used a bi-directional, long short-term algorithm incorporated in a recurrent neural network, which overcomes the vanishing gradient problem related to the recurrent neural network and allows for preserving prolonged periods of contextual information, (3) Third, a central coordinator unit (CCU) assigns an ID to the cloud data center on registration and keeps a copy of cloud data, (4) Finally, a collaborative intrusion detection system (CIDS) consisting of multiple IDS coordinating to detect cyberattacks such as DDoS attacks.

6.11.2. Working

The authors trained and built a recurrent neural network model using a bidirectional long short-term memory algorithm (BiLSTM). During testing, the trained model processes each row of the test dataset (i.e., UNSW-NB15 and BoT-IoT) and classifies them into benign or different types of DDoS attacks such as DDoS TCP, DDoS UDP, and DDoS HTTP.

6.12. NetSprint

6.12.1. General Information

NetSprint [94] detects DDoS attacks by employing semi-supervised learning and model pruning. Semi-supervised learning uses labeled and unlabeled data for training, but there is a larger quantity of unlabeled data. In other words, the tagged data lie approximately on a manifold of much lower dimensions than the input space [95]. NetSprint consists of three modules: (1) the local training module, which focuses on semi-supervised learning; (2) the global aggregation module that applies pruning on the data packets (for instance, fine-grained pruning [96]), which handles the increase in the neural network size, which causes an increase in the memory and computation cost; and (3) the packet parse module, which helps the switch to understand if the received packet has control or traffic information. NetSprint also defines a trustworthy network architecture that consists of (1) the intelligence layer, which is responsible for DDoS attack detection through training; (2) the identification layer, which is responsible for verifying the authenticity of the source address; and (3) the infrastructure layer, which is responsible for truthful storage and computation as we have in a blockchain. For conducting experiments, the authors create a simulated environment using MindSpore.

6.12.2. Working

First, the local training module retrieves the parameters of the already-developed global model. In the next step, the packet parse module retrieves the packet information. The local training module trains the neural-network-based model using locally available information that can comprise labeled and unlabeled data, thus recovering local gradients. The local train module then loads the local gradients to the server, which also obtains inputs from the network. Now, the parse module on the server reads the packet information. The global aggregation module upgrades and prunes the global model, and finally, the system generates the results.

6.13. Blockchain-Based Botnet Detection

6.13.1. General Information

The work in [97] focuses on botnet detection and prevents the escalation of the threat to other IoT devices from the detected botnet. The detection system consists of three parts: descriptor, monitor, and comparator. The descriptor first defines policies for access,

communication, and usage of the IoT device. Then, a policy descriptor and a monitor are responsible for accessing and comparing the device's current state with the defined policies, respectively. However, the authors also describe a collaborative mitigation system, but its description is beyond the scope of our research.

6.13.2. Working

The monitor observes the current state of the devices and calls the comparator for anomaly detection, which confirms the anomalies related to IoT device usage, access, and communication.

Table 4. IoT-based Firmware, Physical, and DDoS Attacks: Classification and detection techniques.

Approach Type	Vulnerability/Attack	Detection System	Technique
Blockchain-based Collaboration	DDoS	Blockchain-based Detection and Collaborative Mitigation System [97], (Sections 6.13 and 7.13)	Policy Violation
		Collaborative Blockchain-based System [78], (Sections 6.3 and 7.3)	If the ratio of the outgoing messages from the busiest node to the second most active node of the system is greater than 2, then the busiest node is a DDoS victim
		Deep Blockchain-based Collaborative Intrusion Detection System [93], (Sections 6.11 and 7.11)	Uses Intrusion Detection Systems trained with bi-directional long short-term memory-based Recurrent Neural Network
	Buffer Overflow, Code Reuse, Replay Attack	Lightweight Collaborative Blockchain-based Anomaly Detection System [77], (Sections 6.2 and 7.2)	Agent identifies the memory region causing failure and passes the information to the user
Blockchain-based Non-Collaborative Systems	Firmware Attack	IoTCop [76], (Sections 6.1 and 7.1)	Monitors inter-message communication between devices and isolates a device not complying with security policy
	DDoS	Framework for Detecting DDoS in an SC-based Blockchain–IoT System [82], (Sections 6.5 and 7.5)	Use of AI techniques in Intrusion Detection System to distinguish network traffic as benign or hacker-based
		AI-enabled System to detect DDoS in Blockchain-based Smart Transport System [79], (Sections 6.4 and 7.4)	Combines autoencoder with Multi-Layer Perceptron to detect DDoS
		Distributed Intrusion Detection System for Blockchain-enabled IoT Network [84], (Sections 6.6 and 7.6)	Integration of Detection System with the Mining pool and use of AI techniques
Non-Blockchain-based Systems	DDoS	NetSprint [94], (Sections 6.12 and 7.12)	Collaborative Learning using semi-supervised learning and model pruning
		IoT-based monitoring System of Banking Sector using Machine-Learning [91], (Sections 6.10 and 7.10)	Justifies SVM for DDoS detection
		Machine-Learning-based Smart Detection System [88], (Sections 6.9 and 7.9)	Smart Detection System works well with Random Forest, XGBoost, and AdaBoost
		Grammar-based filtering and Clustering Algorithm [4], (Sections 6.7 and 7.7)	Detection of suspicious packets and increase in arrival rate
		Machine-Learning-based botnet Detection System [87], (Sections 6.8 and 7.8)	Combined feature Engineering, SMOTE Technology, and Machine-Learning Algorithms

7. Firmware, Physical, and DDoS Attack Detection Techniques

Section 6 discusses the workings and general features of the DDoS detection systems. Here, we discuss their detection techniques.

7.1. Detection Technique Using Security Policy in IoTCop

The work [76] discusses the handling of firmware and physical attacks by IoTCop. Hence, this technique does not help directly detect a device's faulty firmware. Firmware attacks employ rootkit [98] and other malicious software to allow the attacker to control, modify, or replace other IoT devices' OS. Firmware and physical attacks cause a compromised device to inject false commands into the firmware of other IoT devices. In addition, IoTCop monitors the inter-message communication between devices to isolate IoT devices with faulty firmware if the defective device's messages do not comply with security policy.

7.2. Detection Technique Using Control Flow Monitoring in Light-Weight Blockchain-Based Collaborative Model

The model [77] installs a trained IDS agent on the IoT device for anomaly detection. During anomaly detection, the agent interacts with the control server, terminates the application (e.g., `setuid`) through the kernel, restarts it, and monitors its control flow using APIs and libraries. When the application fails, for instance, when the attacker executes an exploit and tries to access the memory region where `setuid` resides, a software vulnerability may cause the `setuid` to fail. The agent will detect the memory region which caused the failure and reveal the problem to the user. `setuid` permits the user to run some programs with privileged permissions.

7.3. Detection Technique Using IoT Agent and SC-Based Collaborative Model

The work in [78] discusses that IoT agents exchange path messages which indicate the traffic information at the associated IoT device. When the length of the path messages exceeds the limit, the last IoT device noticing the exceeding of the path messages, sends the path message to the SC for DDoS detection. The SC calculates the ratio of the outgoing messages from the busiest node, i.e., h_m, to the second most active node of the network such that the ratio is greater than two. If: $\frac{Outbound\ traffic\ from\ h_m}{Outbound\ traffic\ from\ h_i} > 2$, then h_m is a victim of a DDoS attack.

7.4. Detection Technique Using Hybrid Deep-Learning-Based Mechanism for Smart Transport System

The deep learning module combines the autoencoder with a multi-layer perceptron and detects a wide range of DDoS attacks. The authors [79] trained the model recursively by reducing the number of features; this process increases the efficiency of the model.

7.5. Detection Technique Using XGBoost and Random Forest in an SC-Based IoT–Blockchain System

AI techniques such as Random Forest (Section 4.11.1) and XGBoost (Section 4.11.2) help IDS [82] distinguish the incoming traffic as benign or hacker-based. IDS employs correntropy measures to detect the attack by retrieving feature vectors from the incoming traffic and comparing it with feature vectors of the trained dataset.

7.6. Detection Technique Using Distributed Intrusion Detection System (IDS) to Detect DDoS Attacks in a Blockchain–IoT Network

The authors [84] integrate the DDoS detection system with a mining pool. First, the system performs preprocessing, normalizing the features to a specific scale. Next, the detection engine processes the IoT traffic for the detection of normal and abnormal transactions using AI-based Machine-Learning algorithms, i.e., Random Forest (Section 4.11.1) and XGBoost (Section 4.11.2). Finally, in the third phase, normal transactions execute in the mining pool, while the administrator takes care of the abnormal transactions.

7.7. Detection Technique Using Grammar-Based Filtering and Clustering Algorithm

The work in [4] discusses two techniques for DDoS attacks. The first technique indicates the presence of a DDoS attack if the packets captured by the sniffer retrieve some suspicious information, as discussed in the algorithm. However, the authors did not elaborate on what they mean by suspicious information. Their second method indicates that the packets' arrival rate during the DDoS attack would increase, which may affect the service rate. However, the experiment indicates a high service rate even during the attack. The authors cannot measure the maximum system performance due to the limitation in resources.

7.8. DDoS Detection via Botnet Detection Using Machine-Learning in an IoT System

The authors [87] combined several techniques such as feature engineering and SMOTE technology with a couple of Machine-Learning algorithms, namely, KNN (Section 4.11.3), the Gaussian Naive Bayes algorithm, and multilayer perceptron, but found that their model works best with KNN (Section 4.11.3).

7.9. DDoS Detection Using a Machine-Learning-Based Smart Detection System

The authors [88] built their classification model using XGBoost (Section 4.11.2), AdaBoost, Random Forests (Section 4.11.1), and multilayer perceptron, which uses the CICIDS2017 dataset for training. After training, their system can classify the test data into either DDoS attacks or benign ones.

7.10. DDoS Detection Using IoT-Based Monitoring System for the Banking Sector

This work [91] separately trains the IoT-based monitoring system with three Machine-Learning algorithms such as SVM, KNN (Section 4.11.3), and Random Forest (Section 4.11.1) and found that SVM performs the best.

7.11. DDoS Detection Using a Deep Blockchain Framework-Enabled Collaborative Intrusion Detection System

To detect DDoS attacks, the authors [93] developed a collaborative intrusion detection system that employs a recurrent neural network based on BiLSTM utilizing the Keras deep learning library provided by Python for training. The authors divide the dataset into three groups: (1) training, (2) validation, and (3) testing with a 60%, 20%, and 20% ratio, respectively. Implementing their model produces the best accuracy for detecting DDoS attacks.

7.12. DDoS Detection by NetSprint

The authors [94] used an artificial neural network model with 256 hidden layers to train and analyze the network behaviors, including DDoS attack detection. The authors used a CICDDoS2019 dataset to train the model. The training method is collaborative learning, which employs semi-supervised learning and fine-grained pruning, which helps in DDoS detection.

7.13. DDoS Detection using Blockchain-Based Detection and a Collaborative Mitigation System

An IoT device's current state is used to detect a violation using the defined algorithm [97], followed by a comparator to confirm the breach.

8. Future Work and Challenges

8.1. Challenges and Open Issues

8.1.1. Multi-Way Authentication of IoT Devices

One way to handle the weak defenses of IoT devices is to empower them with multi-way authentication. However, manufacturers can equip the IoT devices with multiple password authentications if multi-way authentication is not possible.

8.1.2. Lack of Built-In Mechanism to Stop DDoS

We found that IoT networks and subsystems such as power grid systems and networking protocols do not have any built-in mechanisms to stop DDoS attacks as in Bitcoin blockchains and Ethereum SCs. For instance, network systems can be reset to avoid DDoS attacks based on enormous traffic rates.

8.1.3. Blockchain: Storing Large Files

The blockchain stores transactions in the block. The current block size is 1MB and stores only financial information. Thus, if we want to use blockchain as a database, we must increase the block size. Otherwise, we have to design a technique similar to IPFS (Section 4.7). Hence, future work can focus on finding a solution to this problem.

8.2. Future Work and Suggestions

8.2.1. DDoS Detection and Mitigation: Packet Rate and Rebooting

We propose a two-step technique for detecting and mitigating DDoS attacks. In the first step, we can use SDN to measure the incoming packet rate at all nodes. If the incoming packet rate increases for a node and becomes more significant than the standard rate, it indicates that the node is under a DDoS attack. At that moment, the node can opt for automatic rebooting. Upon startup, the node should change its IP address. Thus, it can save itself from a DDoS attack.

8.2.2. Blockchain: Adoption for Retail

There is no DDoS problem with blockchain [99] because a transaction fee resists the DDoS attack. However, blockchain suffers from the consensus mechanism, which makes the blockchain very slow. Hence, it is not easy to adopt blockchain for retail because instant validation cannot be performed through miners for retail transactions. In addition, the process becomes difficult to adopt if an item has to be returned. Again, we have to apply mining for validating the returning items, which happens often in retail-based transactions. Hence, future work can focus on designing a technique to solve blockchain's adoption problem for retail.

8.2.3. Comparison of Machine-Learning Algorithms

Previous research shows some endeavors involving processing various Machine-Learning algorithms on the same dataset. This sort of research is beneficial because we have a chance to compare the performance of systems based on different algorithms. Moreover, once we know the drawbacks of the tools, the researchers can apply corrective measures. For instance, recurrent neural networks have a problem preserving information for a more extended period. However, we can now use bi-directional long short-term memory to solve these problems. Thus, if we compare the performance of the algorithms, we will identify their problems, and we can solve them.

8.2.4. Federated Learning

Federated learning or collaborative learning is the latest technique for training Machine-Learning-based systems. However, there are different approaches available to implement federated learning. For instance, the method discussed in [100] uses averaging, whereas work in [94] uses pruning for federated learning's implementation. Therefore, we can develop new approaches for its implementation that can support different situations and provide specific benefits.

8.2.5. Collaborative Differential Learning

The work in [82,84] uses XGBoost and Random Forest for training purposes. However, they did not illustrate the reasons for their achievements concerning training the data. Hence, future research can also focus on training the dataset using different Machine-

Learning algorithms called collaborative differential learning and then provide clues for their success or failures in connection with their training procedures.

8.2.6. DDoS Detection and Mitigation: The IoT-FPGA Approach

The work in [101] shows that FPGA devices keep working under attacks. Therefore, we can use an FPGA to configure a circuit and then change the configured circuit to support another one. Based upon this logic, we integrate an FPGA into an IoT circuit or interface an FPGA with an IoT circuit [102]. However, the integrated FPGA has some threat-aware components [103], which would become activated when the IoT network endures a DDoS attack and note down the attackers' IP addresses. Then, a covert link could be used to pass the attackers' collected IP information to a salvaging network, which can administer a DDoS attack on the attackers (Section 4).

9. Conclusions

IoT sensors are responsible for retrieving crucial and sensitive information. This information is then stored in the cloud network and the blockchain. However, both of them can become a victim of DDoS attacks. In this research, we have provided various techniques for detecting DDoS attacks. This can help researchers detect and mitigate DDoS problems. At the same time, researchers can also use these techniques to develop more advanced tools. Our future work will focus on developing a tool for detecting DDoS attacks.

Author Contributions: Data collection, initial draft, Z.A.K.; Supervision, Funding acquisition, A.S.N.; Methodology, structure, Z.A.K. and A.S.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research work is supported by National Science Foundation (NSF) under Grant No: 1821560.

Data Availability Statement: Not applicable.

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

References

1. Smith, G.M. Data Acquisition (DAQ)—The Ultimate Guide. Available online: https://dewesoft.com/daq/what-is-data-acquisition (accessed on 1 November 2022).
2. Thouti, S.; Venu, N.; Rinku, D.R.; Arora, A.; Rajeswaran, N. Investigation on identify the multiple issues in IoT devices using Convolutional Neural Network. *Meas. Sens.* **2022**, *24*, 100509. [CrossRef]
3. Shah, Z.; Ullah, I.; Li, H.; Levula, A.; Khurshid, K. Blockchain-Based Solutions to Mitigate Distributed Denial of Service (DDoS) Attacks in the Internet of Things (IoT): A Survey. *Sensors* **2022**, *22*, 1094. [CrossRef]
4. Ekolle, Z.E.; Kimio, K.; Ryuji, K. Intelligent Security Monitoring in Time Series of DDoS attack on IoT Networks using Grammar base Filtering and Clustering. In Proceedings of the 2018 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), Ishigaki Island, Japan, 27–30 November 2018 ; pp. 37–42.
5. DoS vs. DDoS. Available online: https://www.fortinet.com/resources/cyberglossary/dos-vs-ddos# (accessed on 19 August 2022).
6. Rajan, D.M.; Sathya Priya, S. DDoS mitigation techniques in IoT: A Survey. In Proceedings of the 2022 International Conference on IoT and Blockchain Technology (ICIBT), Ranchi, India, 6–8 May 2022; pp. 1–7.
7. Jing, H.; Wang, J.; Chen, C.L. Detection of DDoS Attack within Industrial IoT Devices Based on Clustering and Graph Structure Features. *Sec. Commun. Netw.* **2022**, *2022*, 1401683. [CrossRef]
8. Salim, M.M.; Comivi, A.K.; Nurbek, T.; Park, H.; Park, J.H. A Blockchain-Enabled Secure Digital Twin Framework for Early Botnet Detection in IIoT Environment. *Sensors* **2022**, *22*, 6133. [CrossRef] [PubMed]
9. Hasan, H.R.; Salah, K.; Yaqoob, I.; Jayaraman, R.; Pesic, S.; Omar, M. Trustworthy IoT Data Streaming Using Blockchain and IPFS. *IEEE Access* **2022**, *10*, 17707–17721. [CrossRef]
10. Azbeg, K.; Ouchetto, O.; Jai Andaloussi, S. BlockMedCare: A healthcare system based on IoT, Blockchain, and IPFS for data management security. *Egypt. Inform. J.* **2022**, *23*, 329–343. [CrossRef]
11. Liu, Z.; Qian, L.; Tang, S. The prediction of DDoS attack by Machine-Learning. In Proceedings of the Third International Conference on Electronics and Communication, Harbin, China, 7 March 2022; p. 6.
12. Acarali, D.; Rajesh Rao, K.; Rajarajan, M.; Chema, D.; Ginzburg, M. Modelling smart grid IT-OT dependencies for DDoS impact propagation. *Comput. Secur.* **2022**, *112*, 102528. [CrossRef]
13. Dai, Q.Y.; Zhang, B.; Dong, S.Q.; Fu, A. A DDoS-Attack Detection Method Oriented to the Blockchain Network Layer. *Sec. Commun. Netw.* **2022**, *2022*, 5692820. [CrossRef]

14. Jiang, S.; Yang, L.; Gao, X.; Zhou, Y.; Feng, T.; Song, Y.; Liu, K.; Cheng, G.; Chen, Y. BSD-Guard: A Collaborative Blockchain-Based Approach for Detection and Mitigation of SDN-Targeted DDoS Attacks. *Sec. Commun. Netw.* **2022**, *2022*, 1608689. [CrossRef]
15. Alduailij, M.; Khan, Q.W.; Tahir, M.; Sardaraz, M.; Alduailij, M.; Malik, F. Machine-Learning-Based DDoS Attack Detection Using Mutual Information and Random Forest Feature Importance Method. *Symmetry* **2022**, *14*, 1095. [CrossRef]
16. Babu, M.R.; Veena, K.N. A Survey on Attack Detection Methods For IOT Using Machine Learning And Deep Learning. In Proceedings of the 2021 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 13–14 May 2021; pp. 625–630.
17. 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]
18. Mittal, M.; Kumar, K.; Behal, S. Deep learning approaches for detecting DDoS attacks: a systematic review. *Soft Comput.* **2022**. [CrossRef] [PubMed]
19. Chaganti, R.; Bhushan, B.; Ravi, V. The role of Blockchain in DDoS attacks mitigation: techniques, open challenges and future directions. *arXiv* **2022**, arXiv:2202.03617.
20. Dalmazo, B.L.; Marques, J.A.; Costa, L.R.; Bonfim, M.S.; Carvalho, R.N.; da Silva, A.S.; Fernandes, S.; Bordim, J.L.; Alchieri, E.; Schaeffer-Filho, A.; Paschoal Gaspary, L.; Cordeiro, W. A systematic review on distributed denial of service attack defense mechanisms in programmable networks. *Int. J. Netw. Manag.* **2021**, *31*, e2163. [CrossRef]
21. Alashhab, A.A.; Zahid, M.S.M.; Azim, M.A.; Daha, M.Y.; Isyaku, B.; Ali, S. A Survey of Low Rate DDoS Detection Techniques Based on Machine Learning in Software-Defined Networks. *Symmetry* **2022**, *14*, 1563. [CrossRef]
22. Eswari, D.S.; P.V.Lakshmi. A Survey On Detection Of DDos Attacks Using Machine Learning Approaches. *Turk. J. Comput. Math. Educ.* **2021**, *12*, 4923–4931.
23. Ashraf, A.; Elmedany, W.M. IoT DDoS attacks detection using machine learning techniques: A Review. In Proceedings of the 2021 International Conference on Data Analytics for Business and Industry (ICDABI), Sakheer, Bahrain, 25–26 October 2021; pp. 178–185.
24. Cheema, A.; Tariq, M.; Hafiz, A.; Khan, M.M.; Ahmad, F.; Anwar, M. Prevention Techniques against Distributed Denial of Service Attacks in Heterogeneous Networks: A Systematic Review. *Secur. Commun. Netw.* **2022**, *2022*, 8379532. [CrossRef]
25. Khan, Z.A.; Namin, A.S. The Applications of Blockchains in Addressing the Integration and Security of IoT Systems: A Survey. In Proceedings of the 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 15–18 December 2021; pp. 2421–2426.
26. Khan, Z.A.; Siami Namin, A. Ethereum Smart Contracts: Vulnerabilities and their Classifications. In Proceedings of the 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 10–13 December 2020; pp. 1–10.
27. Yadav-Ranjan, R.; Brisebois, A.; Banerjee, S. DDoS Attack Identification Utilizing Machine Learning in Circumstances Involving Hacked IoT Devices/Insider Assaults. Available online: https://www.iiconsortium.org/news-pdf/joi-articles/2022-March-JoI-DDoS-Attack-Identification-Using-Machine-Learning.pdf (accessed on 25 August 2022).
28. Goodin, D. One of the Most Powerful DDoS Attacks Ever Hits a Crypto Platform. Available online: https://www.wired.com/story/ddos-attack-botnet-crypto-platform/# (accessed on 24 August 2022).
29. Saad, M.; Njilla, L.; Kamhoua, C.; Kim, J.; Nyang, D.; Mohaisen, A. Mempool optimization for Defending Against DDoS Attacks in PoW-based Blockchain Systems. In Proceedings of the 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Seoul, Republic of Korea, 14–17 May 2019; pp. 285–292.
30. Tulic, A. Is It Possible to Launch a DDoS Attack Using Ethereum's Blockchain? Available online: https://www.quora.com/Is-it-possible-to-launch-a-DDoS-attack-using-Ethereums-blockchain (accessed on 31 August 2022).
31. How the Ethereum Network Handle Ddos Attacks? Available online: https://www.reddit.com/r/ethereum/comments/2iyyk9/how_the_ethereum_network_handle_ddos_attacks/ (accessed on 2 September 2022).
32. Ethereum Network Attacker's IP Address Is Traceable. Available online: https://www.bokconsulting.com.au/blog/ethereum-network-attackers-ip-address-is-traceable/ (accessed on 26 August 2022).
33. Why Is My Node Synchronization Stuck/Extremely Slow at Block 2,306,843? Available online: https://ethereum.stackexchange.com/questions/9883/why-is-my-node-synchronization-stuck-extremely-slow-at-block-2-306-843/9892#9892 (accessed on 26 August 2022).
34. Yang, T.; Liu, Y.; Li, W. Attack and defence methods in cyber-physical power system. *IET Energy Syst. Integr.* **2022**, *4*, 159–170. [CrossRef]
35. Krause, T.; Ernst, R.; Klaer, B.; Hacker, I.; Henze, M. Cybersecurity in Power Grids: Challenges and Opportunities. *Sensors* **2021**, *21*, 6225. [CrossRef]
36. Hu, L.; Wang, Z.; Han, Q.; Liu, X. State estimation under false data injection attacks: Security analysis and system protection. *Automatica* **2018**, *87*, 176–183. [CrossRef]
37. Merlino, J.; Asiri, M.; Saxena, N. DDoS Cyber-Incident Detection in Smart Grids. *Sustainability* **2022**, *14*, 2730. [CrossRef]
38. Maupin, R.Z. What Are Unusual Ports? Available online: https://networkengineering.stackexchange.com/questions/80151/what-are-unusual-ports (accessed on 18 October 2022).
39. Wu, Y.; Weng, J.; Qiu, B.; Wei, Z.; Qian, F.; Deng, R.H. Random Delay Attack and Its Applications on Load Frequency Control of Power Systems. In Proceedings of the 2019 IEEE Conference on Dependable and Secure Computing (DSC), Hangzhou, China, 18–20 November 2019; pp. 1–8. [CrossRef]

40. Yan, S.; Gu, Z.; Park, J.H.; Xie, X.; Dou, C. Probability-density-dependent load frequency control of power systems with random delays and cyber-attacks via circuital implementation. *IEEE Trans. Smart Grid* **2022**, *13*, 4837–4847. [CrossRef]
41. Yan, S.; Nguang, S.K.; Zhang, L. Nonfragile Integral-Based Event-Triggered Control of Uncertain Cyber-Physical Systems under Cyber-Attacks. *Complexity* **2019**, *2019*, 8194606. [CrossRef]
42. Dorato, P. Non-fragile controller design: An overview. In Proceedings of the 1998 American Control Conference. ACC (IEEE Cat. No. 98CH36207), Philadelphia, PA, USA, 26 June 1998; Volume 5, pp. 2829–2831.
43. Seuret, A.; Gouaisbaut, F.; Ariba, Y. Complete quadratic Lyapunov functionals for distributed delay systems. *Automatica* **2015**, *62*, 168–176. [CrossRef]
44. TCP or UDP—Which Protocol Does VoIP Use? Available online: https://www.vipvoip.co.uk/tcp-vs-udp/ (accessed on 28 October 2022).
45. What Is a UDP Flood Attack? Available online: https://www.cloudflare.com/learning/ddos/udp-flood-ddos-attack/ (accessed on 28 October 2022).
46. Sarangam, A. UDP Flooder DDOS ATTACK—A Concise Guide For 2021. Available online: https://www.jigsawacademy.com/blogs/cyber-security/udp-flooder/ (accessed on 28 October 2022).
47. Pal, D. UDP-Based Amplification—The Dangerous DDoS Attack Vector. Available online: https://blog.apnic.net/2022/08/19/udp-based-amplification-the-dangerous-ddos-attack-vector/2022 (accessed on 1 November 2022).
48. Krigman, A. Cyber Autopsy Series: Ukrainian Power Grid Attack Makes History. Available online: https://www.globalsign.com/en/blog/cyber-autopsy-series-ukranian-power-grid-attack-makes-history (accessed on 1 November 2022).
49. What is a Ransom DDoS attack? Available online: https://www.cloudflare.com/learning/ddos/ransom-ddos-attack/ (accessed on 24 August 2022).
50. What is Ransom DDoS (RDDoS)? Available online: https://www.imperva.com/learn/ddos/ransom-ddos-rddos/# (accessed on 24 August 2022).
51. Jawad, A.; Newton, L.; Matrawy, A.; Jaskolka, J. A Formal Analysis of the Efficacy of Rebooting as a Countermeasure Against IoT Botnets. In Proceedings of the ICC 2022—IEEE International Conference on Communications, Seoul, Republic of Korea, 16–20 May 2022; pp. 2206–2211.
52. De Donno, M.; Dragoni, N.; Giaretta, A.; Spognardi, A.; Bugliesi, M. DDoS-Capable IoT Malwares: Comparative Analysis and Mirai Investigation. *Sec. Commun. Netw.* **2018**, *2018*, 7178164. [CrossRef]
53. What Is IPFS (InterPlanetary File System)? Available online: https://moralis.io/what-is-ipfs-interplanetary-file-system/ (accessed on 2 September 2022).
54. What Is Software-Defined Networking (SDN)? Available online: https://www.vmware.com/topics/glossary/content/software-defined-networking.html (accessed on 1 November 2022).
55. Odom, W. Introduction to Controller-Based Networking. Available online: https://www.ciscopress.com/articles/article.asp?p=2995354&seqNum=2 (accessed on 1 November 2022).
56. Contini, A. Software Defined Networking Fundamentals Part 1: Intro to Networking Planes. Available online: https://www.opendaylight.org/blog/2016/11/16/software-defined-networking-fundamentals-part-1-intro-to-networking-planes (accessed on 1 November 2022).
57. Szyrkowiec, T.; Santuari, M.; Chamania, M.; Siracusa, D.; Autenrieth, A.; Lopez, V.; Cho, J.; Kellerer, W. Automatic Intent-Based Secure Service Creation Through a Multilayer SDN Network Orchestration *J. Opt. Commun. Netw.* **2018**, pp. 289–297. [CrossRef]
58. Magyari, A.; Chen, Y. Review of State-of-the-Art FPGA Applications in IoT Networks. *Sensors* **2022**, *22*, 7496. [CrossRef]
59. Jumaa, N. Survey: Internet of Thing Using FPGA. *Iraqi J. Electr. Electron. Eng.* **2017**, *13*, 38–45. [CrossRef]
60. Babaei, A.; Schiele, G.; Zohner, M. Reconfigurable Security Architecture (RESA) Based on PUF for FPGA-Based IoT Devices. *Sensors* **2022**, *22*, 5577. [CrossRef]
61. Machine Learning Terms: Problem with Understanding the Definition of Precision and Recall. Available online: https://www.reddit.com/r/learnmath/comments/wmnzxj/machine_learning_terms_problem_with_understanding/ (accessed on 19 August 2022).
62. Brightlinger, A. Machine Learning: Can't Understand F1 Score Is Harmonic Mean. Available online: https://www.reddit.com/r/learnmath/comments/wnqsw6/machine_learning_cant_understand_f1_score_is/ (accessed on 18 September 2022).
63. XGBoost. Available online: https://www.nvidia.com/en-us/glossary/data-science/xgboost/ (accessed on 22 August 2022).
64. Joby, A. What Is K-Nearest Neighbor? An ML Algorithm to Classify Data. Available online: https://learn.g2.com/k-nearest-neighbor (accessed on 22 August 2022).
65. Doshi, K.; Yilmaz, Y.; Uludag, S. Timely Detection and Mitigation of Stealthy DDoS Attacks Via IoT Networks. *IEEE Trans. Dependable Secur. Comput.* **2021**, *18*, 2164–2176. [CrossRef]
66. Gupta, B.B.; Quamara, M. An overview of Internet of Things (IoT): Architectural aspects, challenges, and protocols. *Concurr. Comput. Pract. Exp.* **2020**, *32*, e4946. [CrossRef]
67. Lombardi, M.; Pascale, F.; Santaniello, D. Internet of Things: A General Overview between Architectures, Protocols and Applications. *Information* **2021**, *12*, 1–20. [CrossRef]
68. Rayes, A.; Salam, S. In *Internet of Things From Hype to Reality*; Springer: Cham, Switzerland, 2017 ; pp. 93–138.
69. IoT Protocols and Their Architecture. Available online: https://www.elprocus.com/iot-protocols-and-its-architectures/ (accessed on 26 August 2021).

70. Liu, H.; Bolic, M.; Nayak, A.; Stojmenovi, I. *Encyclopedia on Ad Hoc and Ubiquitous Computing*; World Scientific Publishing Company: Singapore, 2009; pp. 319–347.
71. Soh, Z.H.C.; Jaafar, A.K.H.A.; Sulaiman, S.N.; Abdullah, S.A.C.; Ibrahim, M.N.; Bakar, A.A. *Fridge Load Management System with AI and IOT Alert*; IOP Publishing: Philadelphia, PA, USA, 2021.
72. Refrigerator, User Manual, SamSung 2021. Available online: https://www.manualslib.com/manual/147316/Samsung-Refrigerator.html (accessed on 19 November 2022).
73. 6LoWPAN From Wikipedia, the Free Encyclopedia. Available online: https://en.wikipedia.org/wiki/6LoWPAN (accessed on 30 October 2021).
74. Business Logic. Available online: https://en.wikipedia.org/wiki/Business_logic (accessed on 26 August 2021).
75. Davis, E. Available online: https://www.practicalecommerce.com/How-Backend-Code-Describes-an-Ecommerce-Business (accessed on 26 August 2021).
76. Seshadri, S.S.; Rodriguez, D.; Subedi, M.; Choo, K.K.R.; Ahmed, S.; Chen, Q.; Lee, J. IoTCop: A Blockchain-Based Monitoring Framework for Detection and Isolation of Malicious Devices in Internet-of-Things Systems. *IEEE Internet Things J.* **2021**, *8*, 3346–3359. [CrossRef]
77. Mirsky, Y.; Golomb, T.; Elovici, Y. Lightweight collaborative anomaly detection for the IoT using blockchain. *J. Parallel Distrib. Comput.* **2020**, *145*, 75–97. [CrossRef]
78. 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]
79. Liu, T.; Sabrina, F.; Jang-Jaccard, J.; Xu, W.; Wei, Y. Artificial Intelligence-Enabled DDoS Detection for Blockchain-Based Smart Transport Systems. *Sensors* **2021**, *1*, 32. [CrossRef] [PubMed]
80. Saripuddin, M.; Suliman, A.; Syarmila Sameon, S.; Jorgensen, B.N. Random Undersampling on Imbalance Time Series Data for Anomaly Detection. In Proceedings of the 2021 The 4th International Conference on Machine Learning and Machine Intelligence. Association for Computing Machinery, MLMI'21, Hangzhou, China, 17–19 September 2021; pp. 151–156.
81. Oliveira, T.P.; Barbar, J.S.; Soares, A.S. Multilayer Perceptron and Stacked Autoencoder for Internet Traffic Prediction. *Network and Parallel Computing*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 61–71.
82. Kumar, P.; Kumar, R.; Gupta, G.P.; Tripathi, R. A Distributed framework for detecting DDoS attacks in smart contract-based Blockchain-IoT Systems by leveraging Fog computing. *Trans. Emerg. Telecommun. Technol.* **2021**, *32*, e4112. [CrossRef]
83. Triebstok, K. How IPFS is Challenging the Web as We Know It. Available online: https://medium.com/innovation/how-ipfs-is-disrupting-the-web-e10857397822# (accessed on 19 August 2022).
84. Kumar, R.; Kumar, P.; Tripathi, R.; Gupta, G.P.; Garg, S.; Hassan, M.M. A distributed intrusion detection system to detect DDoS attacks in blockchain-enabled IoT network. *J. Parallel Distrib. Comput.* **2022**, *164*, 55–68. [CrossRef]
85. Gosavi, A. Tutorial for Use of Basic Queueing Formulas. Available online: https://web.mst.edu/~gosavia/queuing_formulas.pdf (accessed on 16 August 2022).
86. Goodacre, N.; Aljanahi, A.; Nandakumar, S.; Mikailov, M.; Khan, A.S. A Reference Viral Database (RVDB) To Enhance Bioinformatics Analysis of High-Throughput Sequencing for Novel Virus Detection. *mSphere* **2018**, *3*, e00069-18. [CrossRef]
87. Pokhrel, S.; Abbas, R.; Aryal, B. IoT Security: Botnet detection in IoT using Machine learning. *arXiv* **2021**, arXiv:2104.02231.
88. Peneti, S.E.H. DDOS Attack Identification using Machine Learning Techniques. In Proceedings of the 2021 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 27–29 January 2021; pp. 1–5.
89. Mwanthi, D. Getting Started with Recursive Feature Elimination Algorithm in Machine Learning. Available online: https://www.section.io/engineering-education/recursive-feature-elimination/ (accessed on 22 August 2022).
90. scikit-learn: Machine Learning in Python. Available online: https://scikit-learn.org/stable/ (accessed on 22 August 2022).
91. Islam, U.; Muhammad, A.; Mansoor, R.; Hossain, M.S.; Ahmad, I.; Eldin, E.T.; Khan, J.A.; Rehman, A.U.; Shafiq, M. Detection of Distributed Denial of Service (DDoS) Attacks in IOT Based Monitoring System of Banking Sector Using Machine Learning Models. *Sustainability* **2022**, *14*, 8374. [CrossRef]
92. Onel Harrison. Machine Learning Basics with the K-Nearest Neighbors Algorithm. Available online: https://towardsdatascience.com/Machine-Learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761 (accessed on 12 August 2022).
93. Alkadi, O.; Moustafa, N.; Turnbull, B.; Choo, K.K.R. A Deep Blockchain Framework-Enabled Collaborative Intrusion Detection for Protecting IoT and Cloud Networks. *IEEE Internet Things J.* **2021**, *8*, 9463–9472. [CrossRef]
94. Xu, K.; Zheng, Y.; Yao, S.; Wu, B.; Xu, X. NetSpirit: A Smart Collaborative Learning Framework for DDoS Attack Detection. *IEEE Netw.* **2021**, *35*, 140–147. [CrossRef]
95. bok. Semi-Supervised Learning. Available online: https://en.wikipedia.org/wiki/Semi-supervised_learning (accessed on 29 August 2022).
96. Pruning Overview. Available online: https://docs.xilinx.com/r/1.3-English/ug1333-ai-optimizer/Pruning-Overview (accessed on 1 November 2022).
97. Muhammad Sajjad, S.; Rafiq, M.; Yousaf, M.; Aslam, W.; Alshahrani, R.; Nemri, N.; Afzal, H.; Khan, M.; Chen, C.M. Detection and Blockchain-Based Collaborative Mitigation of Internet of Things Botnets. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 1194899.
98. Shacklett, M.E. Rootkit. Available online: https://www.techtarget.com/searchsecurity/definition/rootkit (accessed on 23 August 2022).
99. Gadekallu, T.R.; Pham, Q.V.; Nguyen, D.C.; Maddikunta, P.K.R.; Deepa, N.; Prabadevi, B.; Pathirana, P.N.; Zhao, J.; Hwang, W.J. Blockchain for Edge of Things: Applications, Opportunities, and Challenges. *IEEE Internet Things J.* **2022**, *9*, 964–988. [CrossRef]

100. McMahan, B.; Ramage, D. Federated Learning: Collaborative Machine Learning without Centralized Training Data. Available online: https://ai.googleblog.com/2017/04/federated-learning-collaborative.html (accessed on 1 November 2022).
101. Brasilino, L.R.B.; Swany, M. Mitigating DDoS Flooding Attacks against IoT using Custom Hardware Modules. In Proceedings of the 2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS), Granada, Spain, 22–25 October 2019; pp. 58–64.
102. Zhao, Y.; Cheng, G.; Duan, Y.; Gu, Z.; Zhou, Y.; Tang, L. Secure IoT edge: Threat situation awareness based on network traffic. *Comput. Netw.* **2021**, *201*, 108525. [CrossRef]
103. Elnawawy, M.; Farhan, A.; Nabulsi, A.A.; Al-Ali, A.; Sagahyroon, A. Role of FPGA in Internet of Things Applications. In Proceedings of the 2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), Ajman, United Arab Emirates, 10–12 December 2019; pp. 1–6.