electronics

MDPI

Article

Analysis of Consumer IoT Device Vulnerability Quantification Frameworks

Samira A. Baho and Jemal Abawajy *

Centre for Cyber Resilience and Trust (CREST), Faculty of Science, Engineering and Built Environment, Deakin University, Geelong, VIC 3220, Australia
* Correspondence: jemal@deakin.edu.au

Abstract: The increasing deployment of Internet of Things (IoT) devices in mission-critical systems has made them more appealing to attackers. Cyberattacks on IoT devices have the potential to expose sensitive data, disrupt operations, and even endanger lives. As a result, IoT security has recently gained traction in both industry and academia. However, no research has examined existing IoT vulnerability assessment frameworks in a systematic and comprehensive manner. To address this gap, this paper systematically reviews and analyses the research challenges and state-of-the-art IoT vulnerability assessment frameworks while taking into account both breadth and depth. The study provides insight into current IoT vulnerability assessment approaches, which is useful for ongoing efforts to characterise cybersecurity risks and manage IoT vulnerabilities. It will be of interest to a spectrum of readers, including those in the IoT research community, researchers in cybersecurity, risk and vulnerability management professionals, and others. By offering the latest perspective on the present IoT vulnerability assessment techniques, this study will raise IoT security awareness and facilitate research into IoT vulnerability assessment methodologies. The knowledge provided by this study will also be beneficial to future academics who are interested in the issues and solutions surrounding IoT security. The report also assists in understanding the research direction in IoT vulnerability assessment approaches, making it beneficial for those looking to create new methods for determining IoT vulnerabilities.

Keywords: Internet of Things; IoT security; IoT vulnerability; threats and attacks; vulnerability assessment; vulnerability quantification; systematic literature review (SLR)

check for updates

Citation: Baho, S.A.; Abawajy, J. Analysis of Consumer IoT Device Vulnerability Quantification Frameworks. *Electronics* **2023**, *12*, 1176. https://doi.org/10.3390/electronics12051176

Academic Editor: Antoni Morell

Received: 13 January 2023
Revised: 21 February 2023
Accepted: 21 February 2023
Published: 28 February 2023

1. Introduction

The Internet of Things (loT) is rapidly expanding and attracting widespread interest from a wide range of sectors. The low cost and simplicity of IoT device deployment are the primary driving forces behind the IoT's continued growth. These devices cover a wide range of smart objects, such as voice-activated home assistants (e.g., Alexa), smart locks, smart TVs, and web cameras. The versatility of IoT devices makes them suitable for numerous real-world emerging applications in various areas, such as smart homes [1], health care [2,3], precision agriculture [4], construction industries [5], manufacturing industries [6], and smart grids [7]. It is expected that 29 billion IoT devices will be installed globally by 2030 across industry verticals and other areas [8].

IoT promises ample benefits to consumers and businesses, including convenience, increased efficiency, processes and productivity, service level, and customer satisfaction. While the benefits of IoT are undeniable, IoT devices are riddled with exploitable vulnerabilities [9]. Vulnerability is defined as an inherent flaw in IoT firmware, applications, or services that adversaries could take advantage of in some way to perpetrate an attack. According to a recent study, 70% of IoT devices have security flaws, with each device containing an average of 25 flaws [10]. These vulnerabilities are primarily responsible for a 300% increase in cyberattacks on IoT devices in the first half of 2019 [11]. Adversaries take advantage of IoT vulnerabilities in many ways, such as eavesdropping on consumers' personal information [11,12], creating massive botnets [13], launching distributed denial of service (DDoS) attacks on critical systems [7,14], sending phishing and spam emails [14], and stealing intellectual property worth of millions of dollars [15].

With an increase in IoT vulnerability discovery and exploitation, research into IoT vulnerability detection and mitigation has recently received unprecedented attention. This has resulted in a significant number of published academic articles and survey papers addressing various aspects of IoT security challenges [16–21]. In particular, there is a growing body of literature on IoT vulnerability assessment and quantification frameworks. This is due to the importance of IoT vulnerability assessment in evaluating security risks and developing IoT security strategies. While existing survey papers add to our understanding of IoT vulnerability, they do not address IoT vulnerability assessment and quantification frameworks. Specifically, to our knowledge, no systematic literature review (SLR) on IoT vulnerability assessment frameworks has been conducted. This paper seeks to fill this gap.

Motivated by this, we put forward a state-of-the-art systematic literature review on IoT vulnerability assessment and quantification frameworks. This study covers research published between 2016 and 2022 to identify the common frameworks used to assess and quantify vulnerabilities in the IoT domain. In addition to complementing existing surveys, the SLR underscores the significance of vulnerability assessment frameworks in IoT security. The SLR provides baseline knowledge and serves as a foundation for current IoT vulnerability assessment frameworks. It offers background information for cybersecurity researchers and practitioners seeking to understand current methods and techniques for assessing IoT device vulnerabilities. It also serves as a road map for researchers who want to investigate emerging gaps in IoT vulnerability assessment and quantification or be at the forefront of mainstream research on the subject. Furthermore, the information presented in this paper will be useful in understanding the research direction in IoT vulnerability assessment frameworks, which will contribute to the development of novel techniques for assessing and mitigating IoT vulnerabilities.

Our work advances the state of the art in IoT vulnerability assessment in many ways, including:

- Identifying SLR research gaps on IoT-specific vulnerability assessment frameworks with emphases on vulnerability quantification methodology;
- Presenting a state-of-the-art systematic literature review on IoT vulnerability assessment and quantification frameworks;
- Identifying the key approaches used to assess and quantify security vulnerabilities in IoT, the key analysis techniques, and the primary application domains;
- Identifying limitations of the current IoT vulnerability assessment research so that researchers could address these open challenges;
- Presenting IoT architecture layer-based analysis of IoT security vulnerabilities.

The remainder of the paper is organised as follows: Sections 2 and 3 discuss related work, followed by background information. In Section 4, the research methodology is discussed. The results and discussions are highlighted in Section 5, followed by possible future directions in Section 6. Concluding remarks are in Section 7.

2. Related Work

Security in IoT is a broad research field that covers a wide range of topics. This is evident from many published surveys covering a wide range of IoT cybersecurity issues [16–18,22,23]. The top ten IoT vulnerabilities listed by OWASP (Open Web Application Security Project) [9] reflect the lack of maturity of IoT security vulnerability assessment frameworks. Since a successful exploitation of IoT vulnerabilities potentially jeopardizes privacy, safety, and system availability, IoT vulnerability has recently attracted significant attention from industry and academics [21]. There are a handful of surveys that specifically focus on the growing IoT vulnerabilities. Table 1 summarizes the extant research on IoT vulnerabilities. These surveys exclusively focus on IoT vulnerability identification, detection, and discovery. There is no SLR in vulnerability assessment and quantification frameworks. The SLR presented in this paper addresses these research gaps.

Table 1. Comparative analysis of recent work on IoT security vulnerabilities.

References	Contributions	Challenges Addressed
Mosenia and Jha [16]	A survey on IoT security vulnerabilities and security controls.	Identification of IoT edge layer vulnerabilities.
Neshenko et al. [19]	A comprehensive survey on IoT vulnerabilities and experiment on large-scale exploitations.	Identification of vulnerabilities, attack vectors, impacts, and controls.
Rytel et al. [24]	Survey on publicly available sources of known IoT vulnerabilities.	Identifying information sources of IoT vulnerabilities.
AlLifah et al. [25]	A systematic literature survey of smart home device security vulnerabilities.	Identifying smart home device security vulnerabilities.
Yu et al. [21]	Surveyed research on methods for IoT device vulnerability analysis.	Methods for IoT device vulnerability analysis.
Xie et al. [26]	Survey on IoT firmware security vulnerabilities analysis.	Detecting vulnerabilities in IoT firmware.
Srivastava et al. [27]	Covers a multi-tiered survey covering the causes of IoT-related vulnerabilities across various layers of IoT infrastructure.	Identifying vulnerabilities across various layers of IoT infrastructure.
Anand et al. [12]	Survey covering multiple areas and tools for IoT vulnerability with a sustainability focus.	Identifying tools for monitoring and discovering IoT vulnerabilities.
Meneghello et al. [28]	A survey of security vulnerabilities in real IoT devices and IoT network protocols.	Identifying IoT devices and IoT network protocol vulnerabilities.
Costin et al. [29]	Survey of existing threats and vulnerabilities in IoT devices.	Identifying vulnerabilities in video surveillance, CCTV, and web camera systems.
Nadir et al. [30]	A systematic literature review on security and analysis techniques of IoT firmware.	Techniques of IoT firmware vulnerabilities.
Wright et al. [31]	Reviewed the literature on emulation and dynamic analysis useful for determining if firmware contains security vulnerabilities.	Identifying whether firmware contains security vulnerabilities.
Xie et al. [32]	A brief survey on methods for detecting IoT firmware vulnerabilities and classification.	Detecting IoT firmware vulnerabilities.
Frustaci et al. [33]	Analyzed critical security flaws in the communication and networking protocols used in these layers.	Identifying vulnerabilities in communication and networking protocols.
Qasem et al. [34]	A survey on the detection of software vulnerabilities in embedded devices and firmware.	Detection of vulnerabilities in embedded devices and firmware.
Feng et al. [26]	A survey focusing on the challenges in detecting vulnerabilities in IoT device firmware.	Detection of IoT device firmware vulnerabilities.
Yaqoob et al. [35]	Survey on security flaws in medical devices.	Identifying IoT medical device vulnerabilities.

3. Background

This section presents background information. We will discuss IoT reference architecture and security vulnerabilities from the perspective of IoT architecture.

3.1. IoT System Architecture

Currently, there is no universally agreed-upon established standard of IoT architecture. Figure 1 shows the commonly used IoT architecture, composed of physical, network, and application layers [36–39]. Each layer has its own vulnerabilities and is susceptible to different threats. We map these vulnerabilities. Figure 1 also shows security vulnerabilities related to each layer of the IoT architecture.

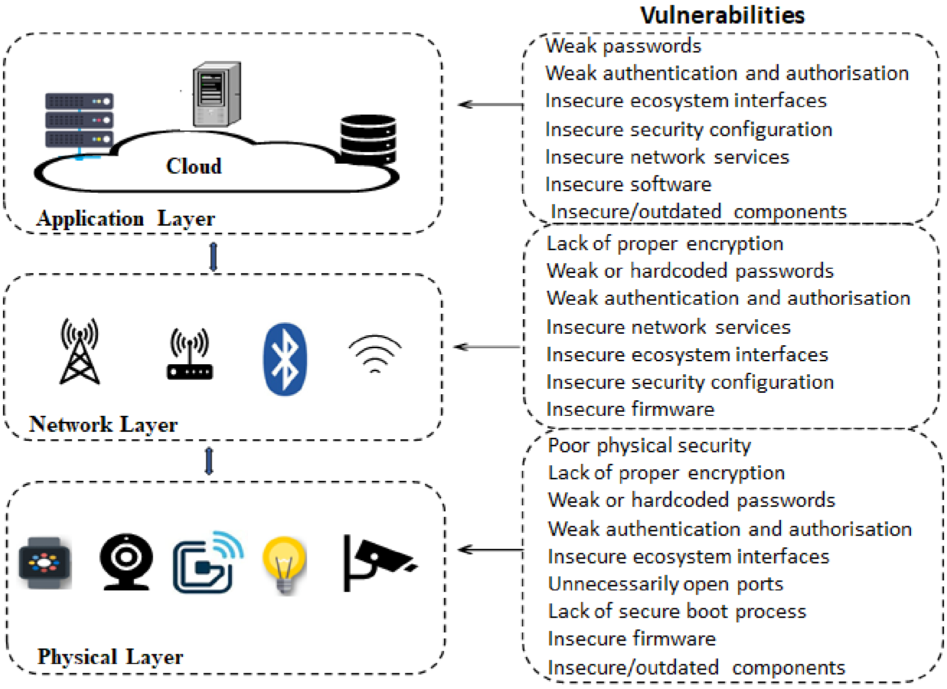
Vulnerabilities
Cloud
Application Layer
Weak passwords
Weak authentication and authorisation
Insecure ecosystem interfaces
Insecure security configuration
Insecure network services
Insecure software
Insecure/outdated components
Network Layer
Lack of proper encryption
Weak or hardcoded passwords
Weak authentication and authorisation
Insecure network services
Insecure ecosystem interfaces
Insecure security configuration
Insecure firmware
Physical Layer
Poor physical security
Lack of proper encryption
Weak or hardcoded passwords
Weak authentication and authorisation
Insecure ecosystem interfaces
Unnecessarily open ports
Lack of secure boot process
Insecure firmware
Insecure/outdated components

Figure 1. Layered IoT architecture with associated IoT vulnerabilities in different layers.

3.1.1. Physical Layer

The physical layer, also known as the perception layer, is the lowest layer of the IoT architecture. It contains all the physical devices, which include IP cameras, Fitbit, sensors, and RFID tags and RFID readers [40,41]. These devices have the capability of sensing the environment and gathering a variety of information, such as motion and temperature, and transmitting the collected information to the network layer. Prevalent security vulnerabilities at the physical layer include lack of physical hardening, lack of proper encryption, weak or hardcoded passwords, weak authentication and authorization, unnecessarily open ports (e.g., open SSH and Telnet ports), lack of secure boot process, insecure firmware, and insecure and outdated components.

Adversaries can leverage weakness in physical security to take control of the device and physically tamper with the devices. For example, attackers can acquire control of the IoT device and modify the Integrated Circuit (IC) and exploit its basic functionality or extract confidential information for remote attack of the IoT system [19]. Moreover, an attacker with physical or wireless access to the IoT device can possibly clone it and use it to infiltrate the IoT network infrastructure [42]. The lack of robust password practices (e.g., using weak, default, and hardcoded passwords) is a major security vulnerability of this layer. The weaker the password, the easier and faster it can be recovered using brute force attacks. Vulnerabilities due to insecure firmware can allow an attacker to insert malicious code into a device, granting the hacker full network access [43].

The absence of a secure boot mechanism results in another IoT device vulnerability. Secure booting is necessary to prevent attackers from compromising an operating system or installing a malicious bootloader into IoT devices. However, IoT devices almost never secure the boot process and do not ensure software integrity [44]. Another IoT vulnerability arises from the lack of proper device update management. This includes a deficiency in firmware validation on the device, an inability to send data securely (they are sent in clear text), a lack of anti-rollback safeguards, and an inability to alert users to security changes brought on by updates. Unauthorized software and firmware updates represent a significant attack surface for IoT devices. By taking advantage of weak update mechanisms, an adversary can launch many attacks [26]. Similarly, an open port that allows access to the IoT device's operating system or other services is a common security flaw at this layer. Attackers employ port scanning to find vulnerable network IoT devices with open ports.

Once such vulnerable devices have been identified, a malicious payload or command is executed in the device.

3.1.2. Network Layer

The network layer's function is to enable device-to-device communication, providing IoT devices access to the Internet, and ensuring efficient information transmission from the perception layer to application interfaces (i.e., to the application layer). Various data transmission technologies, such as Wi-Fi, LTE, Ethernet, BLE, and ZigBee, are used at this layer. Vulnerabilities at the network layer include lack of proper encryption, weak authentication and authorisation, insecure ecosystem interfaces, insecure security configuration, and insecure software/firmware. Adversaries frequently exploit vulnerabilities in insecure interfaces to gain network access. Network layer IoT vulnerabilities mainly arise when attackers exploit weaknesses in IoT communication protocols, such as intercepting sensitive information and launching greater-scale malicious attacks or preventing transmission of legitimate data [45].

Many IoT devices lack encryption capabilities. as demonstrated by the fact that 98% of IoT device traffic on the Internet is unencrypted [46]. Therefore, it is possible for hackers to intercept data sent to and from devices using a Man-In-The-Middle (MITM) attack or other eavesdropping tools to steal login credentials and other sensitive information sent to and from the device. Furthermore, IoT devices that run unnecessary or insecure network services, particularly those connected to the Internet, threaten information confidentiality, integrity, and availability. The lack of attention given to network services security by IoT device manufacturers continues to be a problem for both users and companies.

Default or hardcoded passwords expose connection protocols to cyberattacks [39]. Attackers typically compromise IoT devices by exploiting login credential (usernames/passwords) vulnerabilities and then using the compromised IoT devices to launch large-scale botnets and other malware-related attacks [13,14]. Some 70% of IoT devices are set up to use easily decipherable factory-set default login credentials [47]. Additionally, consumers usually do not change the default access credentials or use simple login credentials. Dictionary attacks based on a list of common login credentials (usernames and passwords) are frequently used by attackers. Specifically, weak, default, and hardcoded login credentials are favoured by attackers for password-guessing exploits. Figure 2 shows the top 10 default and weak passwords used by attackers ('123456' being the most widely used) to compromise IoT devices [48]. These lax password practices pose a significant risk to IoT deployment, allowing attackers to hijack firmware and making IoT devices vulnerable to malware and other cyberattacks. Mirai malware [13], for example, infected IoT devices by successfully logging in with 61 normally used hardcoded default login credentials.

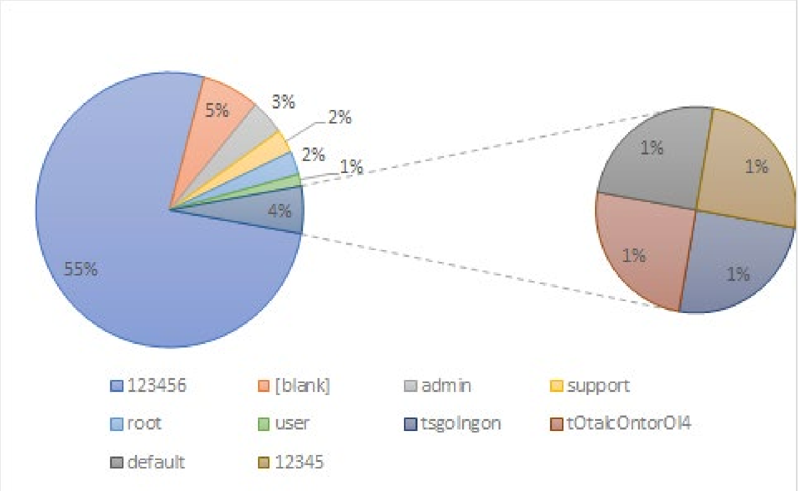

Figure 2. Top ten weak and default passwords in IoT attacks.

3.1.3. Application Layer

The application layer, also called the service layer, is the top layer of the IoT architecture. This layer contains the application user interface and provides the end users with application-specific services, such as integrating and analysing data from IoT devices. Common security vulnerabilities at this layer include weak, guessable, or hardcoded passwords, weak authentication and authorisation, insecure ecosystem interfaces, insecure security configuration, and insecure software/firmware. IoT devices are used in a variety of IoT applications, including smart homes, smart grids, self-driving cars, smart cities, smart buildings, and smart health. Some of the vulnerabilities that adversaries target in IoT applications include unsecured network connections, inadequate and insecure protocols [49], unencrypted data storage, out-of-date IoT application components, weak passwords, inadequate updating methods, and authorisation-related vulnerabilities [38]. Software-related vulnerabilities are exploited by adversaries to compromise the applications and steal or alter sensitive data. Vulnerabilities due to insecure ecosystem interfaces could lead to the compromise of an IoT device or its associated components.

Protocols at this layer such as Message Queuing Telemetry Transfer (MQTT) play a significant role. They provide the backbone of interactions between applications and services running on various IoT devices and on cloud/edge infrastructures [49]. Therefore, the security of the protocols is of the utmost significance. However, each protocol has its different vulnerabilities, thus giving attackers more than one way of intercepting sessions and tampering with data. Therefore, it is quite challenging to guarantee security at this layer due to inherent vulnerabilities, such as the absence of proper security services in the protocols, insecure security configuration, and the misconfiguration of products and services that exposes IoT devices to numerous security risks [49]. For example, by exploiting vulnerabilities in implementation or design of the protocols, an adversary can launch denial-of-service (DoS) attacks at this layer [50]. These are covert attacks, as opposed to large DoS attacks, and they specifically focus on a particular application that the victim is using.

3.2. Vulnerability Assessment

Vulnerability management is a multi-phase process, as shown in Figure 3. The overall aim of the process is to identify, measure, and classify vulnerabilities in systems, applications, and networks based on the risk they represent [51]. Vulnerability assessment is an integral part of the vulnerability management program. It provides visibility to the security weakness of a system and a network and helps prioritize high-risk vulnerabilities that need immediate action.

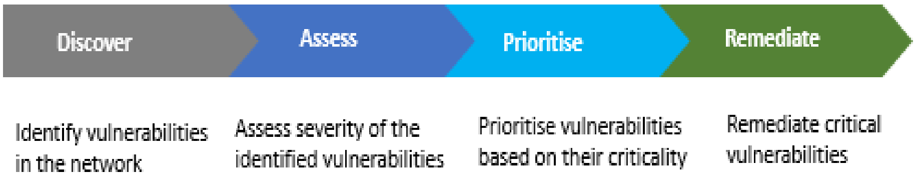
Discover
Assess
Prioritise
Remediate
Identify vulnerabilities in the network
Assess severity of the identified vulnerabilities
Prioritise vulnerabilities based on their criticality
Remediate critical vulnerabilities

Figure 3. Vulnerability management process.

It is not possible to mitigate all identified vulnerabilities effectively; it requires many resources to address every single vulnerability. Furthermore, not all vulnerabilities pose the same risk, thus there is no need to waste resources on mitigating vulnerabilities that do not have major impacts. By assessing vulnerability scores, decision makers would be able to compare a vulnerability's ranking and make a quick decision to focus on critical vulnerabilities that may have serious impacts if exploited.

3.2.1. Vulnerability Assessment Tools

A number of vulnerability scanning and assessment tools, such as Nessus [52] and Shodan [53], are used to automatically detect and assess IoT vulnerabilities. A vulnerability scan searches the network for a set of known vulnerabilities, such as device misconfigurations, missing patches, and open ports. At the end of scanning the network, it returns a list of discovered vulnerabilities. The discovered vulnerabilities are fed into the vulnerability assessment process, which provides organisations with knowledge along with risk scores assigned to vulnerabilities.

Nessus is a vulnerability scanner and assessor that can be used for a variety of purposes. It is commonly used to test vulnerabilities in a variety of devices and applications. Moreover, it provides a robust set of vulnerability assessment options, in addition to the ability to discover several vulnerabilities that include default logins, vulnerable SSH, service misconfiguration, unpatched operating systems, and software that could be exploited by attackers. It also uses the Common Vulnerability Scoring System (CVSS) [54] to assign risk level ('Critical', 'High', 'Medium', 'Low' and 'None') to the detected vulnerabilities.

Shodan [53] is widely used to gather information about consumer IoT devices connected to Internet. The tool identifies information on devices, such as the service name and version, hostnames, domains, geographic location, organisation, and operating system, among other things. It deploys a banner-based method to identify and collect information on a variety of device categories with accessible IP addresses, including webcams, routers, refrigerators, fish tanks, printers, traffic lights, smart TVs, and industrial control systems (ICS). Shodan uses a searchable database to maintain the information discovered on the devices. It provides access to the database via a web interface or an API. It also includes a set of filters for extracting tailored information from the database, such as the device location (country), the device names, the operating systems that run on the device, and the services and ports. Furthermore, it has a feature that identifies exploit code for services provided by discovered IoT devices [55]. Shodan is, regrettably, also a preferred tool of adversaries for carrying out reconnaissance. A hacker, for example, can discover Internet-connected ICS device information using Shodan, such as IP addresses, available open services, and known vulnerabilities, and launch an attack on the ICS.

Shodan and Nessus are used in combination sometimes. For example, Shodan is used to gather information about IoT devices, which is subsequently evaluated by Nessus to see if there are any potential vulnerabilities and to report their scores.

3.2.2. Quantifying Vulnerabilities

Once vulnerabilities are identified, it is important to quantify them for the purpose of determining their score. A number of resources for tracking and cataloguing vulnerabilities in consumer software and hardware exist. These resources include the Common Weakness Enumeration (CWE) system [56], the National Vulnerability Database (NVD) [57], and CVSS system [54]. NVD is a public repository where the list of known vulnerabilities is maintained, whereas CVSS is an open and free de facto vulnerability scoring system.

CVSS generates scores for vulnerabilities based on three groups of metrics: base, temporal, and environmental. Each of these metric groups has a set of metrics that captures aspects of a vulnerability. The base metric describes the basic properties of a vulnerability that remain unchanged over time and in the user environment. It is made up of two types of metrics (i.e., exploitability metrics and impact metrics). In contrast, the temporal group represents vulnerability attributes that change over time and not across user environments. Finally, the environmental group is used to represent vulnerability traits that are specific to a user's environment. Each metric is assigned a value by a scorer such as a vulnerability analyst.

The base metrics group is used to generate a score between 0 and 10. A score of 0 is the lowest and 10 is the highest score. The base score can be modified by assigning values to the temporal and environmental metrics. The vulnerability score is used to understand the threat level of a vulnerability and to prioritise responses and resources to high-scoring vulnerabilities.

4. Research Methodology and Materials

This study aims to investigate state-of-the-art IoT vulnerability assessment frameworks with emphases on vulnerability quantification while considering both breadth and depth.

To achieve this objective, we adopted the guideline drawn from Kitchenham et al. [58]. This systematic methodology is composed of three phases, as outlined in Figure 4.

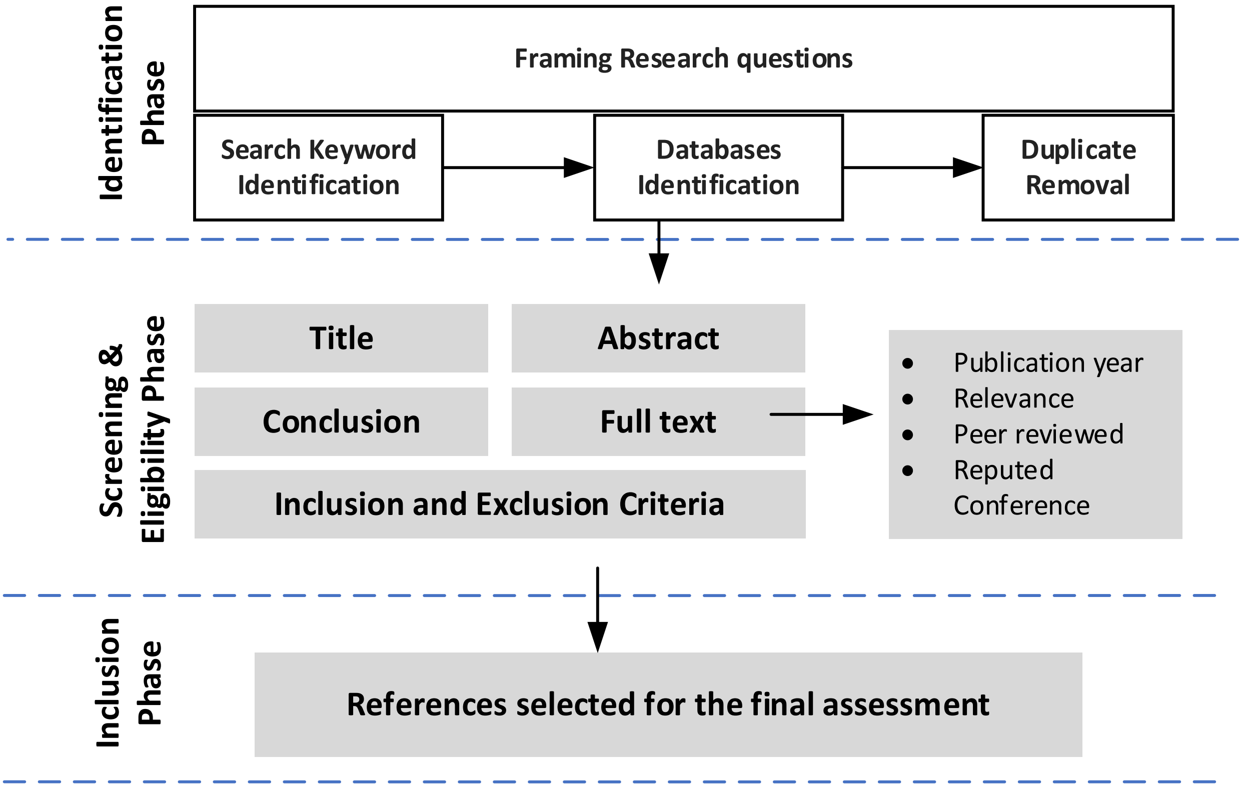

Figure 4. Three phases of a systematic literature review process.

4.1. Identification Phase

In this phase, we formulated the research questions and identified both the search keywords and the databases to be searched.

4.2. Formulation of Research Questions

This study will address the following research questions (RQs):

- RQ1: What are the trends in cybersecurity IoT vulnerability assessment and quantification research?
- RQ2: What are the key frameworks used to assess and quantify potential security vulnerabilities in IoT?
- RQ3: What are the primary application domains used to demonstrate the utility of the approach?
- RQ4: How are vulnerability quantification methods validated?

4.2.1. Digital Database Information Sources

A number of electronic databases were considered. Table 2 displays a representative list of the databases searched, along with their corresponding URLs. Several factors contributed to the selection of these database information sources. These databases are the most widely used databases in similar studies. They are reputable information sources to search for scholarly journal papers and conference articles. Furthermore, these databases were chosen due to the fact that they contain materials that are peer-reviewed. To avoid publisher bias, we started by selecting papers from Google Scholar.

Table 2. Research source databases.

Digital Databases	URL
Google Scholar	www.scholar.google.com (accessed on 13 January 2023)
IEEE Xplore	www.Ieeexplore.ieee.org (accessed on 13 January 2023)
Springer	www.link.springer.com (accessed on 13 January 2023)
ACM Digital Library	www.dl.acm.org (accessed on 13 January 2023)
Science Direct	www.sciencedirect.com (accessed on 13 January 2023)
Scopus	www.scopus.com (accessed on 13 January 2023)

4.2.2. Search Keywords

To search the above electronic databases, we formulated search strings. We constructed the keywords for searching the databases from our research questions based on altered PICOC criteria, as in [59]. With the help of the criteria, we were able to formulate the following keywords: Internet of Things, vulnerability assessment, vulnerability scoring, risk identification, risk assessment, and CVSS.

4.3. Screening and Eligibility Phase

Figure 5 shows the search process flowchart diagram, which is based on Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) [60].

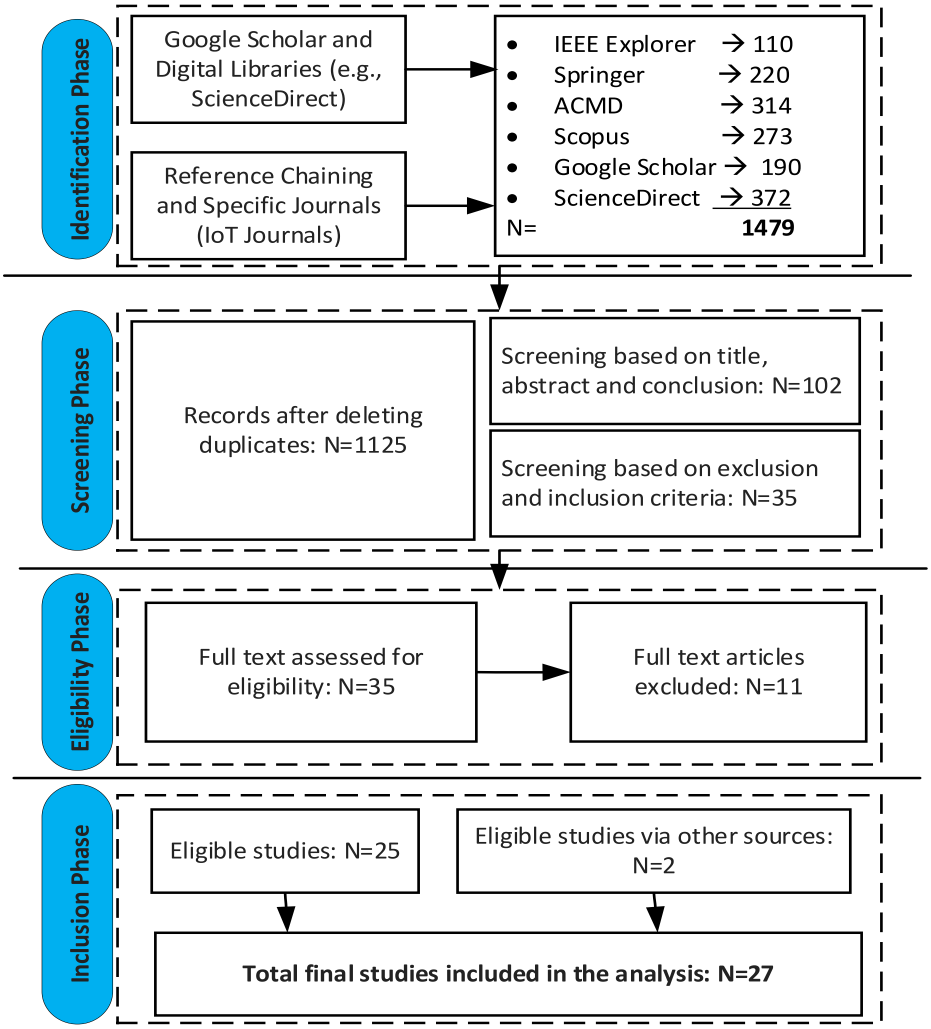
Identification Phase
Google Scholar and Digital Libraries (e.g., ScienceDirect)
Reference Chaining and Specific Journals (IoT Journals)
IEEE Explorer → 110
Springer → 220
ACMD → 314
Scopus → 273
Google Scholar → 190
ScienceDirect → 372
N= 1479
Screening Phase
Records after deleting duplicates: N=1125
Screening based on title, abstract and conclusion: N=102
Screening based on exclusion and inclusion criteria: N=35
Eligibility Phase
Full text assessed for eligibility: N=35
Full text articles excluded: N=11
Inclusion Phase
Eligible studies: N=25
Eligible studies via other sources: N=2
Total final studies included in the analysis: N=27

Figure 5. Visualization of the search process flowchart.

The initial search resulted in 1479 papers. We further filtered the papers from the initial query through multiple phase processes, as shown in Figure 5. In the first step of our search process, each selected database was queried independently based on search strings and with basic inclusion and exclusion criteria. Step two is to obtain unique articles for review. To achieve this, we removed duplicate records returned from the queried databases. Step three is to further filter papers from step two by reviewing the title, abstract, and conclusion. Step four is to filter papers based on full inclusion and exclusion criteria. The final step is to critically go through the remaining papers by reading the full text and selecting studies that are relevant to our systematic literature review.

4.3.1. Inclusion and Exclusion Criteria

We adopted the following inclusion criteria for the purpose of deciding the relevance of the identified articles: (i) the articles are published in the chosen databases to maintain and ensure research quality; (ii) the articles focus on IoT vulnerability assessment in IoT environments; (iii) the articles use vulnerability score quantification; (iv) the articles must be accessible from the database; (v) the articles must be written in the English language; and (vi) the articles must be peer reviewed.

The following papers were not further considered: (i) all papers that did not meet the above-stated inclusion criteria; (ii) all papers without IoT vulnerability quantifications; (iii) the full text of the articles is not available or accessible; and (iv) dissertations (i.e., MS and PhD theses), technical reports, and short papers of less than three pages.

4.3.2. Quality Assessment

To assess the quality of the papers, we used the quality assessment method described in [61]. We developed a checklist of questions for judging the quality of the papers to ensure that the studies included contributed significantly to the SLR. Table 3 below outlines our checklist questions:

Table 3. List of questions.

Q #	Question
1	Are the objectives of the research clearly stated?
2	Does the study cover the answers to our questions?
3	Does the paper clearly explain the proposed/used method?
4	Is the proposed method validated?

After downloading articles from the selected digital libraries, the authors independently reviewed the papers from the refined search lists to decrease selection bias and decided which papers should be included or excluded. A final set of papers that focused on IoT vulnerability assessment and that met the inclusion and eligibility criteria was included in our literature review.

4.4. Data Extraction and Synthesis

The primary goal of this phase was to generate data extraction forms to keep accurate information gathered from primary studies [62]. The data extraction for the selected studies was primarily guided by the form shown in Table 4. The items under the column labelled "Data extracted" in Table 4 were chosen to correspond with the research questions and objectives.

The data extraction and synthesis process were carried out by thoroughly reading each of the selected studies and extracting relevant data. EndNote and Excel were used to manage the data, which included the following columns: paper ID (used to uniquely identify each paper), authors, paper title, publication date, publication source (IEEE, Elsevier, etc.), type of paper (Journal, book chapter, conference, etc.), assessment methods (e.g., CVSS, Nessus, Game theory, Graph, Tree, etc.), validation methods (e.g., simulation, penetration test, testbed, etc.), and description of the application domain to find out in which area the study was used ((e.g., critical infrastructures, smart home, smart health). By carefully examining the full text of each primary study, the necessary data were extracted and synthesised with the intention of including discursive analysis for a variety of issues concerning IoT vulnerability assessment and quantification frameworks. The final set of assessed articles is shown in Table 5.

Table 4. Data extraction for each study.

Data Extracted	Explanation
Paper ID	Unique identifier of the paper
Bibliographic references	Authors, title, type, year, and publication source
Data collection method	Example: Shodan, CVE, experiment, observation, etc.
Assessment methods	CVSS, Nessus, Game theory, Graph, Tree, etc.
Validation methods	Simulation, penetration test, testbed, etc.
Context of application domain	Description of the study domain (e.g., critical infrastructures, smart home, smart health)

Table 5. Final set of reviewed articles.

ID	References	Year	Publisher	Type	Methods	Application	Validation
A1	[55]	2022	ACM	J	CVSS	Electronic commerce	Testbed
A2	[63]	2021	Elsevier	J	Graph/Tree	Healthcare	Simulation
A3	[64]	2019	IEEE	C	CVSS	Smart home, Healthcare, Consumer IoT, Electronic commerce	Numerical
A4	[65]	2021	IEEE	C	Nessus	IP camera	Testbed
A5	[66]	2020	Springer	C	Graph/Tree	Internet of Industrial Things	Numerical
A6	[67]	2021	arXiv	O	CVSS	Healthcare	Numerical
A7	[68]	2016	IEEE	C	CVSS	Internet of Industrial Things	None
A8	[69]	2019	Elsevier	J	Graph/Tree	IoT networks	Simulation
A9	[70]	2020	Elsevier	J	CVSS-Ext	Consumer IoT devices	Numerical
A10	[71]	2022	MDPI	J	CVSS	Smart home, critical infrastructures	Testbed
A11	[72]	2017	IEEE	C	Nessus	Healthcare	Scanning
A12	[73]	2017	IEEE	C	Nessus	Consumer IoT devices	Scanning
A13	[74]	2022	Elsevier	J	CVSS	Cyber-physical system	Simulation
A14	[75]	2018	IEEE	C	Nessus	Smart home	Scanning
A15	[76]	2021	IEEE	C	CVSS-Ext	Smart home	Simulation
A16	[77]	2020	Springer	J	CVSS-Ext	Internet of Industrial Things	Simulation
A17	[78]	2016	IEEE	C	CVSS-Ext	Electronic commerce	Scenarios
A18	[79]	2022	IEEE	C	CVSS-Ext	IoT smart grid	Simulation
A19	[80]	2021	O	J	CVSS	Smart home	Simulation
A20	[81]	2018	IEEE	J	Graph/Tree	Internet of Industrial Things	Simulation
A21	[82]	2018	IEEE	J	Graph/Tree	Internet of Industrial Things	Simulation
A22	[83]	2019	Elsevier	J	Graph/Tree	None	Simulation
A23	[84]	2018	Elsevier	J	Game theory	Smart home	Scenario
A24	[85]	2019	IEEE	C	Others	Smart home	Simulation
A25	[86]	2022	MDPI	J	Others	None	Simulation
A26	[87]	2020	arXiv	O	Others	None	Scenario
A27	[88]	2017	Elsevier	J	CVSS	Smart home, wearable healthcare	Numerical

5. Results and Discussion

In this section, we present the results of the systematic review process. The analyses are conducted in the context of the RQs. First, we will discuss the analysis of research trends in IoT vulnerability assessment. Vulnerability assessment frameworks will be discussed in Section 5.2 The IoT application domain used and the validation methods are described in Sections 5.3 and 5.4, respectively.

5.1. Analysis of Research Trends in IoT Vulnerability Assessment

We examined original published articles over seven calendar years to determine the trends in cybersecurity IoT vulnerability assessment and quantification (see RQ1).

Based on the included studies in the review, Figure 6 shows the trend in IoT vulnerability assessment framework research by year. The result shows an upward trajectory in IoT vulnerability assessment publications. The increase in publication on the topic of IoT vulnerability assessment is in line with the increasing attention recently paid to IoT vulnerability by the security community. In fact, several attacks, such as the Mirai attacks, [13,14] have caused enormous revenue losses and, as a result, identifying IoT device-level vulnerabilities [64] has emerged as a serious challenge. Similarly, the quantity of literature review papers published in recent years [16,19,21,24–35,65,89] corroborates the increasing research interest in IoT vulnerability.

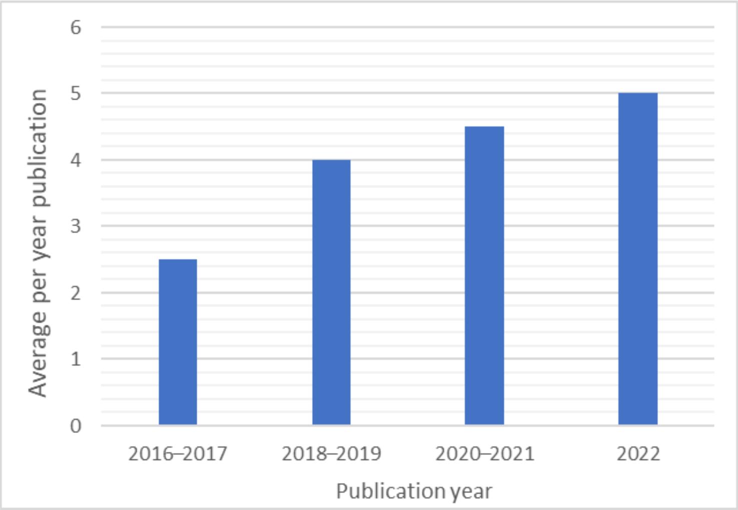

Figure 6. Distribution of the selected papers by year.

Figure 7 shows the trends in research publication by type. Journal publication is 11% higher than conference publication. Conference publication per year is roughly 1.6, while journal publication is almost two per year. There are two articles that appear in arXiv. In fact, the earlier publications were all conference publications, while publications in journals are getting more attention towards 2020. In 2020 and 2022, the number of published papers increased significantly. This increase could be attributed to the fact that IoT cybersecurity threats continue to grow year after year.

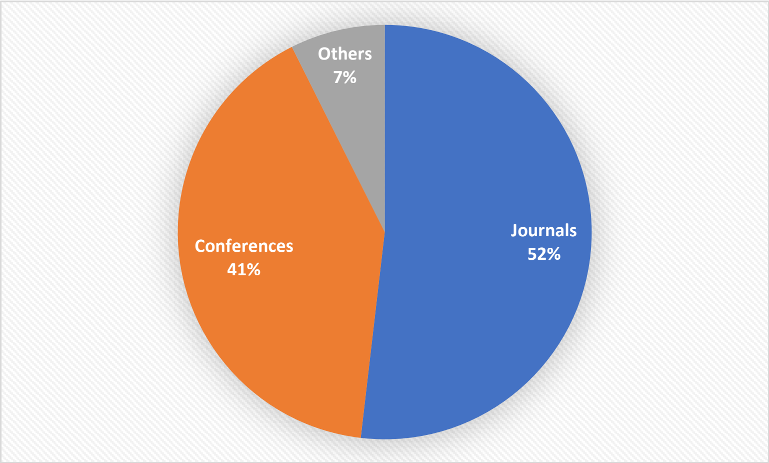

Figure 7. Distribution of the selected papers by type.

Figure 8 shows the trends in research publication by the publishers. IEEE published 45% of the studies, followed by Elsevier at 22%. Figure 9 shows the distributions of the studies by type and publisher. IEEE studies are 83% conference papers and 17% journal papers. Given that both ACM and Springer have conferences comparable to those of IEEE, it is surprising that they have lagged way behind IEEE. Another observation we made is that open access publisher MDPI published an equal number of journals as IEEE. Elsevier was the top in terms of journal publications.

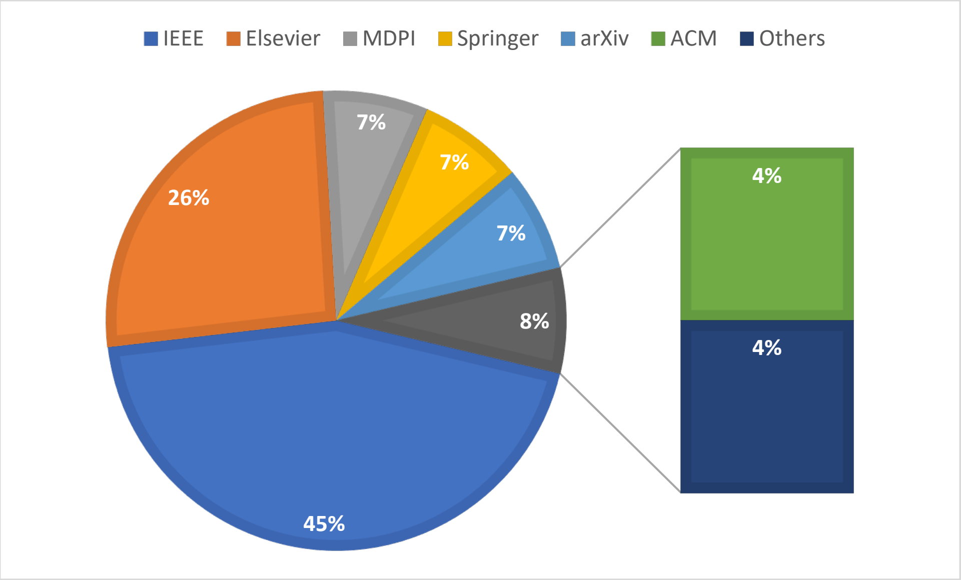

Figure 8. Distribution of the selected papers by publisher.

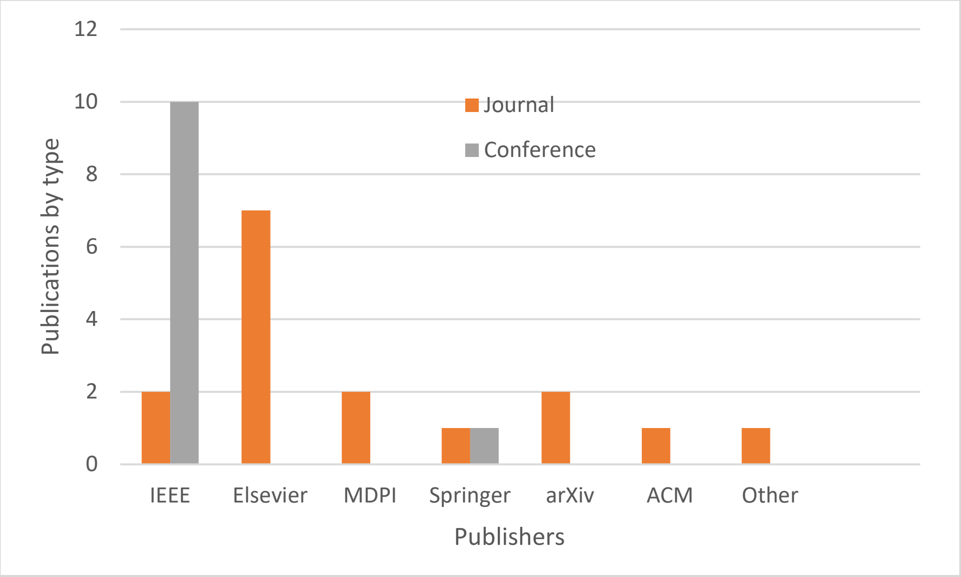

Figure 9. Distribution of the selected papers by type and publisher.

Overall, despite growing evidence that hackers are becoming more interested in IoT, research on IoT vulnerability assessment and quantification is still in its early stages. The number of research papers in IoT vulnerability assessment and quantification is quite small in comparison to the number of IoT devices in use today. Currently, research has mainly concentrated on IoT vulnerability identification and discovery, with a dearth of literature on IoT vulnerability assessment and quantification. Knowing that vulnerabilities are half of the equation in IoT security, identified vulnerabilities must be assessed and quantified. Therefore, there are significant research opportunities to close gaps in this area.

5.2. Vulnerability Assessment Frameworks

A vulnerability assessment is a systematic review of security weaknesses in a system and an evaluation of the risk of their consequences, and it is an integral part of a cyber risk management program. To minimise cyberattacks in IoT, organizations need to conduct 360-degree vulnerability assessments; therefore, there is an urgent need for research on new approaches that calculate the true score that vulnerability poses to an organisation. The number of vulnerability quantification methods proposed in the literature has increased in tandem with the number of active attacks on IoT networks [66]. The aim of the second research question (RQ2) is to determine the vulnerability score assessment methods used in the articles. Vulnerability assessment frameworks are valuable tools for highlighting specific types of cyberthreats. There are several frameworks and we identified six of them in the studies. Table 6 summarizes the vulnerability assessment methods.

Table 6. Vulnerability score assessment methods.

Methods	Remarks	References
CVSS	IoT vulnerabilities are assigned scores using the CVSS scoring system.	[55,64,67,68,71,74,80,88]
Nessus	Vulnerability scanners are used to automatically score vulnerabilities.	[65,72,73,75]
CVSS Extension	Specialized versions of the original CVSS are modified to fit a particular environment or application scenario.	[70,76–79]
Graph/Tree-based	Methods based on attack graphs and attack trees are used to score vulnerability	[63,66,69,81–83]
Game theory	Game theory is used to score the IoT vulnerabilities.	[84]
Others	Quantify vulnerability without relying on conventional methods such as CVSS and Nessus.	[85–87]

Figure 10 shows the overall distribution of the six vulnerability assessment frameworks used by the studies. An important observation is that the CVSS system [54] is the most widely used framework for vulnerability assessment and quantification. It is used by 30% of the studies to quantify vulnerabilities. This is closely followed by CVSS with an extension (CVSS Extension) framework at 22%. Graph/Tree-based frameworks are used by 18% of the studies, followed by Nessus and New frameworks at 15% each. Game theory-based vulnerability assessment and quantification frameworks are the least used in the studies analysed in this paper. While the CVSS scoring system was not designed specifically for IoT systems, it is widely used by researchers to quantify IoT vulnerabilities, or indirectly through the Nessus tool. Interestingly, CVSS-based frameworks (original, extended, and Nessus-combined) account for approximately 70% of vulnerability assessment frameworks. In the following subsections, we will highlight each class of frameworks in detail.

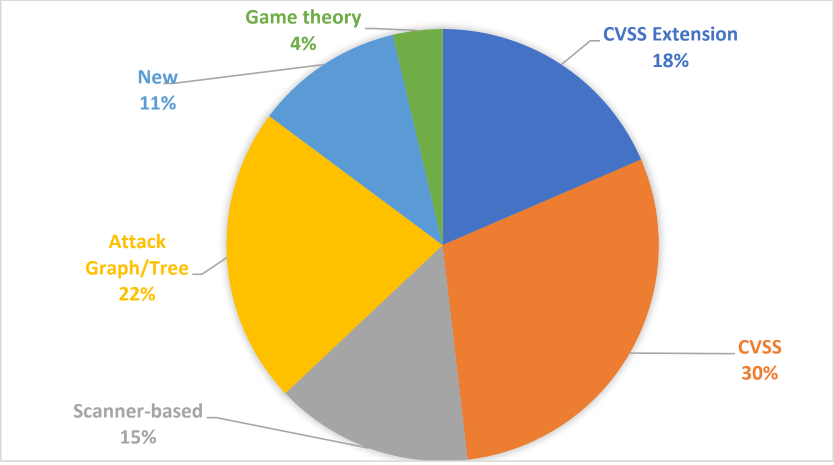

Figure 10. Distribution of vulnerability assessment frameworks.

5.2.1. CVSS-Based Frameworks

The framework discussed in Oser et al. [55] is focused on the estimation of present IoT device risks and predicting future risk based on the currently installed firmware version vulnerabilities. The authors developed a framework called SAFER that uses a CVSS 3.0 value of existing unpatched vulnerabilities for specific IoT device models. The current risk of IoT devices is estimated using the aggregate CVSS 3.0 value of all vulnerabilities in an IoT device. SAFER also predicts future security issues that may arise from the device based on previous vulnerability information and IoT device patching intervals. This is a complex framework, and it will not work in the absence of a prior history of vulnerabilities in specific device models and information on how quickly those vulnerabilities were patched. In addition, IoT firmware analysis possesses several challenges, chief of which is accessibility of firmware images for IoT devices. IoT device firmware images are proprietary, and often tend to be encrypted. Moreover, analysing the firmware image is error-prone because it requires many manual activities such as assembling vulnerability, exposing content. The firmware of the device may also include third-party libraries. It is quite common for vendors of the IoT device not to document these libraries as relevant to their IoT devices.

Rizvi et al. [64] seek to identify IoT device-level vulnerabilities and calculate the security level of IoT devices based on the identified vulnerabilities. The authors analysed IoT devices for common vulnerabilities and used the CVSS base metric group (exploitability, impact, and scope metrics) to determine quantitative and qualitative scores for each vulnerability on an individual IoT device. They averaged the individual vulnerability scores of each device to generate an overall IoT device score. The issue with this study is that it used a row CVSS score for the vulnerabilities without taking into account factors that may influence IoT vulnerabilities. Therefore, understanding device-level vulnerabilities and determining the security level of a specific device is critical for their acceptance.

Allouzi and Khan [67] developed a framework for identifying and assessing vulnerabilities in the Internet of Medical Things (IoMT). The framework is based on a Markov chain and utilizes CVSS as well. They examined selected IoMT vulnerabilities that enable unapproved privileges as well as threats that can exploit these flaws to get unauthorized access to the IoMT network. They calculated scores for the vulnerabilities by defining CVSS information based on the IoMT network. They also used the Markov transition probability matrix to compute the probability distribution of IoMT security threats. The authors do not say much about the CVSS score that was used and why it is used, and the work does not provide insight into which vulnerabilities should be prioritised for mitigation and security measures.

Ando et al. [68] proposed a theoretical security requirement and risk assessment framework. The framework is based on the 5W-tree (who, what, when, where, and why) and fault-tree analysis. The 5W-tree is used to represent all relevant threats, whereas fault-tree is used for analysis of all causes of the threats. The model also uses CVSS v3 base metrics for risk analysis. To analyse the risk, the IoT vulnerability scores from CVSS are mapped to the 5W-tree and the risk values are computed using a data flow diagram (DFD). The major issue with this approach is that it must be reconstructed for different connected car models. This study lacked formal analyses as well as real-world case studies.

Akhilesh et al. [71] address device-level vulnerability analysis. Each IoT device is assigned a score depending on the number of vulnerabilities detected in the device. Specifically, each vulnerability in the device is assigned a CVSS base score. The values of the vulnerabilities in the device are then aggregated to determine the device's score. Since this study only takes into account the five most prevalent vulnerabilities in IoT devices, the size of vulnerabilities taken into consideration is insufficient.

Jiang and Atif [74] proposed a machine learning (ML)-based vulnerability assessment framework with vulnerability scoring capability. Given reported vulnerability instances with no assigned scores, the ML-based method automatically assigns CVSS scores to the vulnerability instances. A majority voting system is used to address CVSS score compatibility issues. This method streamlines vulnerability investigation while reducing the possibility of mistakes due to manual error.

Yadav et al. [80] presented a client-server framework that is based on target graphs for analysing IoT devices and networks. The aim is to discover the possible ways a hacker can exploit vulnerabilities and compromise the target system. Basically, target graphs are attack graphs, except that target paths begin at an arbitrary node and attempt to reach the target node, whereas attack paths begin at the target node and explore all possible sources of infiltration. For each vulnerability associated with a node in the graph, the CVSS score is used to score the criticality level. The framework uses the NVD database to detect vulnerabilities based on the state of IoT nodes. As a result, zero-day vulnerabilities will be missed. Furthermore, it is not clear if the framework can be applied to IoT sensors/actuators in addition to gateways/aggregators. This is due to the fact that IoT sensors lack the necessary resource capacity to install IoT-PEN.

Ge et al. [88] described a conceptual five-phase framework that can be used to assess potential attacks on IoT devices. The five phases are data processing, security model generation, security visualization, security analysis, and model updates. The framework is based on the Hierarchical Attack Representation Model (HARM). CVE is used to identify vulnerabilities, and the CVSS system is used to score vulnerabilities.

5.2.2. Scanner-Based Frameworks

In this subsection, we highlight studies that use vulnerability scan and vulnerability assessment.

Vulnerability assessments using vulnerability scanners such as Shodan and Nessus discover and assess vulnerabilities and assign criticality scores [65,72,73,75]. Biondi et al. [65] proposed a framework that combines penetration testing and vulnerability assessment to analyse cybersecurity threats to smart connected cameras. The approach first conducts penetration testing using open-source tools such as Nmap on the smart cameras. The results of the penetration testing are then fed into Nessus to score the vulnerabilities discovered in the penetration testing phase.

McMahon et al. [72] used the combination of Shodan and Nessus to discover vulnerable medical devices on the Internet and to analyse and score vulnerabilities, respectively. The same approach was applied in Williams et al. [73], where scanning for susceptible Internet-connected consumer IoT devices was done by Shodan and the results were fed into Nessus to analyse and assign scores to the detected vulnerabilities. Bugeja et al. [75] wanted to find exposure of smart connected cameras to cyberthreats on a global scale, with vulnerability scanning of the smart connected cameras followed by vulnerability assessment of the vulnerable smart connected cameras. For vulnerability scanning, Shodan was used to automatically collect data on smart connected cameras. To identify security vulnerabilities and assign scores, the authors relied on the CVE database to retrieve publicly disclosed vulnerabilities. The relevant data elements (e.g., the vendor's name and firmware version) from the Shodan records were scanned against the CVE database to identify CVE IDs for vulnerabilities. The authors were able to discover the exact location of the cameras, firmware version, open ports, and so forth.

5.2.3. Extended CVSS-Based Frameworks

The CVSS system uses a set of parameters and equations to calculate vulnerability scores. Several researchers [70,76–79] have highlighted issues that could arise when using the default CVSS for the IoT environment and proposed ways to ameliorate the shortcomings in CVSS scoring.

Rizvi et al. [70] identify data privacy as a core element in IoT vulnerability quantifications, but the standard CVSS system does not account for privacy. The authors then go on to extend CVSS with attributes for recognising Transparency, Unlikability, and Intervenability (TUI) to capture IoT device-related data privacy risks when scoring vulnerability impacts. Given that a large portion of the data collected by IoT devices is classified as Personal

Identifiable Information and personal health information (PHI), considering privacy as a factor in computing vulnerability scores is sound. Note that the CVSS system is mainly concerned with data security issues in IoT devices when calculating vulnerability scores by analysing confidentiality, integrity, and availability.

Duan et al. [76] discussed a security assessment framework for IoT networks that uses machine learning (ML) and two-layer graphical security modelling. The vulnerability assessment model predicts the exploitability metric (EM) and impact metric (IM) from NVD using ML and NLP (natural language processing). The predicted EM and IM are then fed into a graphical security model. The top layer of the graph consists of an attack graph and is used to show network connectivity. The bottom layer consists of an attack tree for each node in the network and is used to show vulnerability data. The predicted EM and IM are used to compute the CVSS score. The framework creates likely attack routes from connectivity and vulnerability data of each network node (including the vulnerability scores) as inputs, and then analyses network security using security metrics and the attack route information.

Ur-Rehman et al. [77] proposed a vulnerability assessment framework that customizes CVSS for hybrid environments (i.e., IoT networks and IT). The authors identified the shortcomings of CVSS v3 and developed a framework called CVSS_{IoT} for industrial control systems. In the framework, the attack vector and the attack complexity vector parameters value are modified, and a new vector called the human safety index (HI) is introduced to capture the IoT-related context. The attack vector is assigned higher numeric weights because IoT devices are easier to physically or locally access than conventional IT devices. In contrast, the attack complexity vector is assigned lower numeric weights on the assumption that, compared to traditional IT devices, attacking an IoT node requires specific knowledge. The idea of considering unique characteristics of IoT devices when determining scores for the vulnerabilities is novel. However, there is no reason given as to why three parameters are enough to capture IoT device characteristics. In addition, there is no discussion as to how the numerical values assigned to the parameters are derived.

Qu and Chan [78] emphasise that using a CVSS base score to assess IoT system vulnerabilities in a BLE wireless network can yield inaccurate results. The authors extended the original CVSS scoring equation by introducing a set of factors relevant to a BLE-based IoT system security framework. They specifically broaden the authentication metric to include newer security variables intrinsic to Bluetooth technology. This is due to the inability of the formulae to account for the influence of BLE wireless network specifics. To address this shortcoming, the authors used an approach based on Bayesian probability to extend the CVSS base score to account for all relevant BLE wireless network factors when estimating base scores.

Rashed et al. [79] discussed a vulnerability assessment framework for an IoT smart grid. The framework quantifies vulnerabilities by calculating a CVSS score based on attack likelihood and erroneous predictions. The framework considers a number of variables, including the likelihood of an attack, how quickly an attack spreads from a parent node to child nodes, and how well the metering system works.

5.2.4. Graph-Based Frameworks

Several IoT vulnerability assessment and scoring methods that leverage attack graphs/trees have been put forward as well [66,69,81–83]. Attack graphs and attack trees are two graphical techniques that are frequently used to model how a system could be attacked. They are an efficient way to model attack scenarios in cybersecurity as well as to quantify security vulnerabilities. These approaches generate a graph or tree for IoT networks, with devices acting as nodes and interactions between devices acting as edges. They are essentially used to show collection attack paths an attacker could take in order to achieve his/her goals.

Stellios et al. [63] addressed the challenges of vulnerabilities that arise from interactions between components of a cyber–physical system (CPS). A framework for assessing IoT devices induced interaction vulnerabilities between IoT systems and other systems in a CPS, with emphases on identifying attack paths. The framework is built around qualitative and quantitative weights, attack trees used to show vulnerability data, a recursive algorithm to construct all the attack paths, and CVSS exploitability metrics to assess the vulnerability of each interaction and each attack path.

Yang et al. [66] describe a security method for evaluating the vulnerability exposure of an IoT device. The relationship between system vulnerabilities and their following nodes is represented by a state attack graph. A successful node exploitation indicates the likelihood of success from one vulnerability to the next. The chance of this occurring is then assessed to enhance the precision of the related risk.

Yiğit et al. [69] proposed an approach to determine IoT network security level metrics based on CVSS scores for hardening the network. The framework uses a compact and cost-aware attack graph-based model. The algorithm calculates the success probability of a node vulnerability using the CVSS base metric (i.e., exploitability score). The graph is traversed backwards from the vertices representing the attacker's objectives to obtain the set of attack paths. Each link is assigned a cost and time weight ranging from 0 to 5 to mitigate node vulnerabilities. Based on these weight values, each link in the attack graph is assessed and given a new score. This method identifies the least expensive and most time-effective vulnerable paths. These routes are preferred when implementing vulnerability mitigation strategies. The main problem with this model is that arbitrary weights for ranking cost and time are assumed, but forecasting these arbitrary weights is difficult because they vary from person to person and industry to industry [77]. Furthermore, the complexity of the algorithm for backwards graph traversal to obtain all of the attack paths from the attack graph grows exponentially with increases in the node numbers.

George and Thampi [81] addressed the security issues in Industrial IoT (IIoT) networks that arise from the vulnerabilities present in the IIoT devices. Specifically, the authors focus on the relationships among the vulnerabilities and proposed an IIoT risk assessment security framework based on a weighted and directed attack graph. The framework captures the relationships between the various vulnerabilities in IIoT networks. The framework assigns scores to the vulnerabilities using CVSS base metrics. The CVSS scores are assigned to the edges as the edge weights.

Wang et al. [82] discussed a vulnerability assessment framework for IIoT that relies on the formation of an attack graph. The framework uses CVSS in conjunction with attack graphs and maximum loss flows to evaluate risk in IIoT networks. Specifically, the framework creates an attack tree from the attack graph for each vulnerability, and a maximum flow algorithm is applied to the tree to calculate the attack risk for various attack paths. The attack risks are measured by vulnerability scores derived from the basic metrics, the time metrics, and the environment metrics to form the CVSS.

George et al. [83] discussed a multi-attacker and multi-target graphical framework for edge computing and IoT vulnerability assessment. The framework is used for identifying vulnerability-based cyberattacks and their interrelations. To evaluate risk in edge computing devices and apply suitable strategies to mitigate, the framework needs to contain hackers, targets, and vulnerability relationships in the network. The framework employs CVE to find flaws and the CVSS system to score them.

5.2.5. Game Theory-Based Frameworks

Game theory has been widely used to model different sorts of cyberattacks and has been recently used in IoT vulnerability assessment and quantification [84]. Game theory enables theoretical analyses of cyberattacks and provides cybersecurity professionals with useful insights in designing security countermeasure strategies.

Lee et al. [84] developed a vulnerability quantification framework based on Game theory. The approach is based on a static Game theory with imperfect information and a constant attacker strategy. To quantify attacks and security activities, the technique employs an attack tree. The overall approach is staged as (i) modelling of the game strategy, (ii) the cost-impact analysis, and (iii) the computing payoff. The cost of an attack and the security measures are determined in the cost-impact analysis, and the impact of each measure is measured based on the estimated costs and attack risk. The effect of the act on the systems is then computed using the features of the intended system. A rating between 0 and 10 is assigned to the risk of an attack activity based on a common vulnerabilities and exposures score (CVE).

5.2.6. New Frameworks

Frameworks that do not rely on the standard CVSS system for vulnerability assessment and quantifying are referred to as new frameworks [85–87].

Payne et al. [85] discussed a framework that uses vulnerability data from IoT devices and their underpinning parts as well as SIEM logs collected from such devices' communications and operations activities. It is based on a flow network for measuring the security of an IoT network and a given IoT device within the network. The flow network, which the authors refer to as an attack circuit, aids in the modelling of potential attack paths as well as the assessment of the security state of the represented IoT network and each individual device within it. An approach for computing a vulnerability score based on the IoT device composition, IoT device CVEs, and IoT device behaviour is described. An attack graph is created in such a manner that, even when IoT devices do not communicate on the network, an attack route may exist between them. The problem with this approach is that the attack circuit complexity rises exponentially as the devices in the network increase. Ntafloukas et al. [86] discussed a risk assessment framework for cyber–physical systems (CPS). The strategy is based on threat sources' cyber–physical properties, security weaknesses, and types of physical impact. The level of risk is determined using a decision scale and importance indices. The framework is of interest to interested parties who are attempting to integrate the cyber domain into their system's risk assessment processes.

Shojaeshafiei et al. [87] describe a fuzzy inference system that quantifies network and IoT device vulnerabilities using goal-oriented metrics. The framework requires a questionnaire filled out by security experts before quantifying vulnerabilities. The authors created a set of questions based on an organization's security requirements as goals (a numerical security vulnerability score of each security factor) that can be traced back to a sequence of quantifiable questions. Based on the security requirements extracted from the questionnaire, the authors mapped security questions to NIST SP800-53 and ISO 27001 security standards. They then used fuzzy logic for vulnerability quantification. The disadvantage of the framework proposed in [87] is that it requires a questionnaire to be completed by security professionals to construct the quantification process. In summary, the advantage of these approaches is that they can be applied to other areas as well.

5.3. IoT Application Domain

We now consider the primary application domains used to demonstrate the utility of the IoT vulnerability assessment approaches (RQ3). Table 7 shows the various practical applications used by the researchers to demonstrate their vulnerability assessment approaches. We categorized the application domains used to demonstrate the utility of the IoT vulnerability assessment approaches into five classes, as shown in Table 7.

IoT technology is used in many sectors, such as healthcare, business, and home automation. Figure 11 shows the distribution of IoT application domains used in the assessment frameworks studied. The majority of studies focused on a single IoT application domain, with few studies considering multiple IoT application domains. The smart home is by far the most widely used IoT application followed by IIoT. The reason for the majority of home-based IoT research is because it serves as the basis for IoT deployments. Figure 11 also shows a significant number of studies on critical infrastructures. This is due to a recent increase in ICS cyberattacks. These studies provide insights helpful to understand the broad critical infrastructure vulnerability and threat landscapes. In the following subsections, we will briefly look at each application domain.

Table 7. Application domains used in the articles.

Category	Remarks	References
Smart home	This includes home automation, smart devices such as smart hubs, smart bulbs and smart connected cameras.	[63–65,70,71,75,76,80,84,85,88]
Healthcare	Applications in the healthcare area such as medical devices, including pacemakers, physician programming devices, activity tracking devices.	[64,67,70,72,88]
Consumer IoT devices	This category includes IoT devices such as RFID devices, near-field communication, webcams, smart TVs, and printers	[55,66,70,73]
Electronic commerce	These includes IoT-enabled shopping carts and IoT in supply chain systems	[64,70,77,78]
Industrial Internet of Things (IIoT)	This class includes smart grids, industrial control systems, and smart transportation systems.	[68,69,74,79,81–84,86,88]

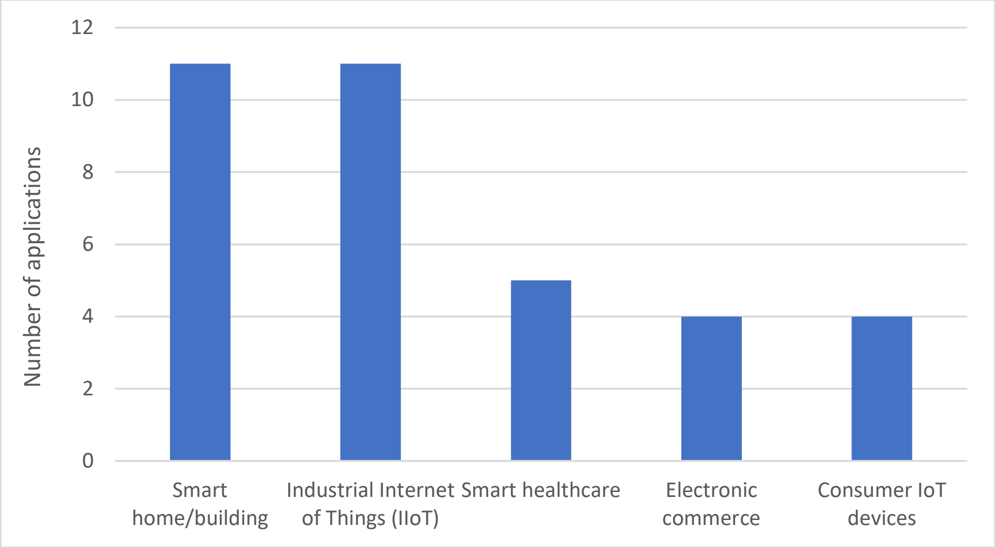

Figure 11. Distribution of IoT application domains used in the assessment frameworks.

5.3.1. Smart Home/Building

At the moment, a growing number of studies are focusing on smart home devices, home automation, and smart devices such as smart hubs, smart bulbs, and smart connected cameras [63–65,70,71,75,76,80,84,85,88]. There is no universally accepted definition of a smart home, but there is a general understanding that smart homes contain networked IoT devices that communicate with one another seamlessly to provide services such as local and remote control of the home environment, which includes heating, lighting, cooling, and so on; health monitoring; and independent/assisted living arrangements. Many studies have used smart homes or smart buildings as case studies to evaluate and demonstrate the frameworks.

Rizvi et al. [64] analysed threats and attacks that could be initiated via a web camera and smart hub product vulnerabilities. Biondi et al. [65] investigated exposure of a specific model of web camera to cyberthreat. Yadav et al. [80] used the Phillips Hue bulbs system, whereas Akhilesh et al. [71] attempted to discover IoT device vulnerabilities in smart home settings. Both Rizvi et al. [70] and Bugeja et al. [75] sought to discover the exposure of smart Internet-connected cameras as well as the sensitive, potentially private data they collect to cyberthreats. Smart camera systems are increasingly being installed in homes, offices, and cities, with the aim to improve citizen security. Hackers through Mirai malware [13] were able to use thousands of devices, including many smart cameras, to establish a botnet. Given such exploitation, as well as the very sensitive nature of the data that these IoT devices make use of, it is intuitive to evaluate their vulnerability to cyberthreats.

Duan et al. [76] used smart buildings as an application that included resource tracking subsystems that use wearable IoT devices connected to the burglar alarm. Some of the

IoT devices considered in the smart building include sensors for CO detection and smoke detection, thermometers, and alarms such as fire alarms and burglar alarms. The case study in Lee et al. [84] used a smart home area network (HAN) as an instance of a social IoT network. The HAN is composed of various home IoT appliances and smart meters. Payne et al. [85] demonstrate the usefulness of their approach in the context of a smart home with IoT devices such as an Amazon Echo Dot and a Belkin WeMo smart plug. Ge et al. [88] considered a smart TV that is connected to Wi-Fi and a tablet equipped with a ZigBee that runs Android. They investigated the possibility of compromising the smart home through exploiting vulnerabilities in these devices.

5.3.2. Smart Healthcare

Smart healthcare is another application area that is clearly seen in the studies [63,64,67,70,72,88]. Smart healthcare refers to the use of Internet of Medical Things (IoMT) devices and applications such as wearable devices (e.g., blood pressure monitors, smart watches, and smart contact lenses) to allow seamless patient care and monitoring.

Stellios et al. [63] used vital systems (e.g., near-patient infusion pumps) and services (e.g., online remote healthcare services) as a test case. Their goal was to identify vulnerabilities that may arise from a wide range of potential interactions between all devices, networks, and interfaces. Rizvi et al. [64] investigated threats and attacks that could be launched through vulnerabilities in three healthcare devices: pacemakers, physician programming devices (PPD), and activity tracking devices (ATD). Allouzi and Khan [67] concentrated on IoMT devices, such as wearable devices that form the IoMT edge network (i.e., Wireless Body Area Network (WBAN)). Rizvi et al. [70] evaluated privacy threats due to Abbott pacemaker vulnerabilities. McMahon et al. [72] identified several Internet-enabled medical devices that included wireless insulin pumps, wireless MRI scanners, wireless pacemakers, glucose monitors, as well as manufacturer-specific devices from Omron Corporation and Welch Allyn. Ge et al. [88] considered WBAN in the context of wearable healthcare monitoring and described attack scenarios.

5.3.3. Electronic Commerce

The IoT presents unprecedented opportunities and risks for businesses. Therefore, the studies also used a variety of electronic commerce applications and devices [64,70,77,78]. Rizvi et al. [64] used e-commerce domain technologies, such as near-field communication (NFC) and radio frequency identification (RFID). Both RFIDs and NFC are susceptible to many cyber risks, including payment processing fraud, eavesdropping, and replay attacks [90]. For Rizvi et al. [70], privacy challenges that POS terminal and radio-frequency identification (RFID) tag vulnerabilities pose were the main objective for considering these devices. Ur-Rehman et al. [77] used IoT devices embedded in a smart supply chain system to manage temperature-sensitive items (e.g., perishable items). Qu and Chan [78] used a shopping cart IoT system that was connected to a BLE wireless network.

5.3.4. Industrial Internet of Things

The Industrial Internet of Things (IIoTs) is used in a variety of critical infrastructure sectors, including energy, water, manufacturing, and transportation. Various applications in these areas are also prominently featured in the studies [68,69,74,79,81–84,86,88].

Ando et al. [68] used connected car systems as a case study to demonstrate their conceptual framework. Yiğit et al. [69] employed IoT networks, such as those used in critical industrial control environments. Jiang and Atif [74] used cyber–physical systems (CPSs) devices that included Remote Terminal Units (RTUs), Master Terminal Units (MTUs) and Human Machine Interfaces (HMIs). Ge et al. [88] used wireless sensor networks (WSNs), where the sensor nodes in a WSN monitored the environment and collected temperature and humidity data from the environment.

Rashed et al. [79] used the IoT smart grid application as a use case to examine their framework. George and Thampi [81] used an IoT solar network that was interconnected with an industrial plant control system. Wang et al. [82] focused on measuring the vulnerability of the Industrial Internet of Things in order to ensure the confidentiality and integrity of data transmission. George and Thampi [83] used a smart airport as a test case to demonstrate vulnerability graph modelling, risk assessment, and mitigation strategies. Ntafloukas et al. [86] used a cyber–physical system with emphases on an IoT-enabled bridge as a use case. An IoT-enabled bridge is a bridge embedded with a wireless sensor network (WSN). It is a critical infrastructure in the transportation domain, and the WSN component is vulnerable to cyberattacks.

5.3.5. Consumer IoT Devices

Consumer IoT devices such as smart TVs are becoming more common. Given the potential implications of loT device vulnerabilities, it is critical to understand the underlying security risks these devices pose. Classes of IoT devices are also identified in the studies [55,66,70,73]. Oser et al. [55] considered general IoT devices present on large-scale enterprise networks. The IoT devices are fetched from networks using network scanning tools. Yang et al. [66] examined consumer IoT devices such as web cameras and smart phones. Rizvi et al. [70] examined the impact of vulnerabilities in consumer IoT devices, such as the Fitbit Surge and Amazon Alexa, on privacy. Williams et al. [73] analysed consumer IoT devices, including smart TVs, smart watches, webcams, and printers.

5.4. Validation Methods

Different approaches are used to validate the IoT vulnerability assessment frameworks. We now discuss how vulnerability quantification methods are validated (RQ4). Figure 12 shows the distribution of the validation methods in the studies. Simulation is by far the most common validation method used in the studies. There is no actual implementation of the frameworks or lage-scale studies. The lack of data is a major challenge in validating vulnerability quantification frameworks.

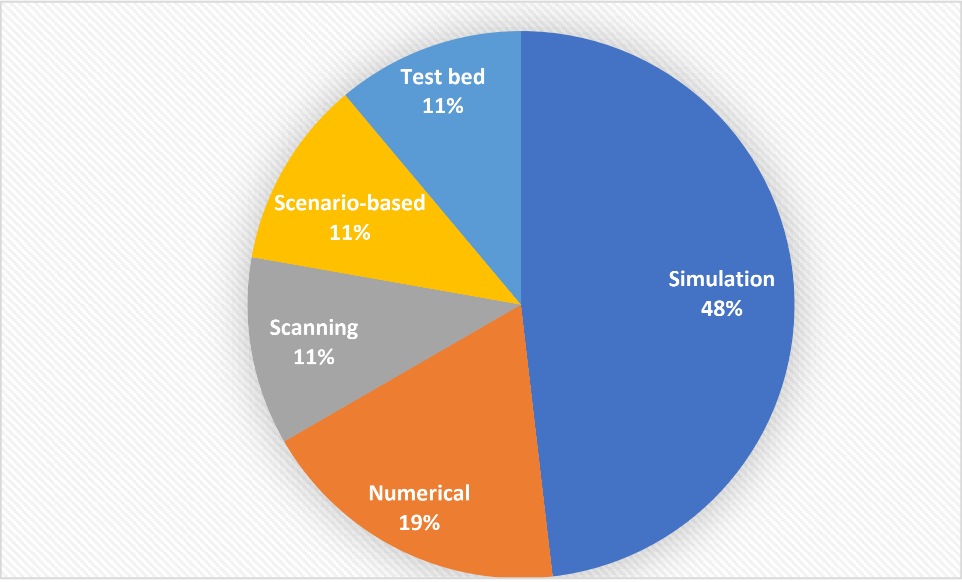

Figure 12. Validation methods for IoT vulnerability assessment approaches.

5.4.1. Simulation

As can be seen from Figure 8, 64% of the methods are validated through simulation [63,69,76,77,79–83,85,86]. Yiğit et al. [69] used simulation on Microsoft Windows 7 Ultimate with Service Pack 1 to analyse the complexity and cost analysis of their framework. Various attacks graph setups were employed in the simulation, each with a varying number of initial states, exploits, and edges. However, the attack graph model used in the experiment is on a small scale, with only a few IoT devices. Furthermore, no consideration was given to the unique characteristics of IoT devices, such as heterogeneous protocols, mobility, physical proximity, and so on.

Stellios et al. [63] validated the effectiveness and accuracy of the framework by simulating real-world scenarios within smart homes and hospital settings. In the simulation, IoT devices, such as infusion pumps, smart lamps, thermostats, and IP surveillance cameras, as well as personal computers, network routers, and access points, were used in both the smart home and the hospital. Several well-known CVEs based on previous research were assigned to each device. The results show that the framework is effective at identifying and assessing cyberattack paths.

Duan et al. [76] evaluate their framework with a case study based on a smart building system. The system consists of a variety of smart sensors and several subsystems (lighting, audiovisual, security, fire detection, maintenance, resource tracking, and HVAC subsystems). Each subsystem provides unique functions and is in charge of carrying out a portion of the smart building's functionalities. The Light Gradient Boosting Machine (LGBM) is used to predict the metrics in the CVSS base score as well as the different types of privileges. All predicted scores for privilege are more than 95%, while scores for integrity and exploitability metrics are more than 90%. The predicted scores for confidentiality are around 90%, whereas the predicted scores for availability are around 83%. The system model considers the IoT devices as unmovable once installed. However, IoT devices such as wearable devices in a realistic environment are frequently movable. The authors also assume all IoT devices can maintain connection always, which may not be the case in real environments.

Ur-Rehman et al. [77] used simulation to evaluate the framework's validity with a case involving a supply chain system. They used an in-house-developed tool known as a vulnerability security analyser (VSA) for simulation. The simulated system model consists of the standard IT infrastructures such as database servers as well as IoT sensors and client gadgets. The authors compared the performance of the framework against actual vulnerabilities from the NVD. The results confirmed that CVSS_{IoT} assigns more realistic numeric weights to IoT nodes by analysing the unique characteristics of an IoT system.

Rashed et al. [79] validate their framework through simulation by considering the IEEE-300 bus of the smart grid. False data injection attacks (FDIA) were used to assess the likelihood of attack success and their spread throughout the system. A variety of attack cases were examined using dynamic estimation and plain metering data. For example, the CVSS score was calculated using the likelihood of attack and incorrect estimates.

Yadav et al. [80] used simulation on a Ubuntu Linux machine. The authors evaluated the model's scalability and performance using metrics such as runtime consumed, node count, attack pathways created, and node flaws. They evaluated their framework on various network topologies but provided no information on the IoT devices used.

George and Thampi [81] simulated an IoT solar network connected to a control network for an industrial plant as a use case and evaluated it using various security parameters. For this use case, a collection of vulnerabilities prevalent in various IIoT network devices were taken into consideration, along with their normalised base scores. Simulation was conducted using many synthetic graphs, with various structures and sizes, and the results were presented. It would be interesting to compare their framework with other similar techniques and analyse the overhead of the framework.

Wang et al. [82] simulated a physical server that hosts a database with three virtual machines. The simulation also has an adversary that intends to steal the database's root login information on the physical server in order to obtain business data. To assess the vulnerability, the cost and reward of an attack is computed for two cases. The first case is on each side of the attack map. The second case is on various attack paths. This enables the network's possible critically vulnerable path to be determined. The findings suggested that when assessing network vulnerabilities, the attacker's characteristics should also be considered. Despite the fact that their proposal is for an IIoT network, they do not evaluate their framework in the context of a proper IIoT scenario.

George and Thampi [83] used simulation to validate their framework. A smart airport is modelled with an edge-based IoT network and cloud computing data centre. The edge–IoT network is composed of a set of IoT sensors and monitors. The system is regarded as having a number of known vulnerabilities, including ones that allow remote code execution, code injection, and IoT device default support credentials. The test was run under the presumption that there are numerous attack paths that lead to edge computing servers, which attackers could exploit to their advantage to execute system commands and gain access to system-level privileges. The results demonstrate that the framework is capable of assessing risk at network edge computing devices.

Payne et al. [85] employ vulnerability data from NVDs and IoT device network traffic statistics to quantitatively test the practical utility of their strategy in the setting of a smart home. An assortment of networks and device activity metrics were used. The authors used three small networks to demonstrate the effectiveness of their framework. This is primarily because of the complexity of the attack circuit. Two of the networks consist of one IoT device with corresponding CVE and one network with two IoT devices with all their related CVEs.

Ntafloukas et al. [86] used Monte Carlo simulations to model and evaluate a practical case study of an IoT-enabled bridge exposed to a cyberattack. The framework was evaluated using four attack scenarios relevant to cyber–physical systems. The attacks were based on the use of various control barriers capable of preventing and detecting cyber–physical attacks. The authors reported that 76.6% of simulated case scenarios are ranked as high-risk. Control barriers that operate in both physical and virtual space can decrease cyber–physical risk by 71.8%.

5.4.2. Numerical and Empirical

Rizvi et al. [64] presented vulnerability scores for three application domains: healthcare, commerce, and home. The authors tested each IoT device with a selected small number of known vulnerabilities, such as login credentials. CVSS scores are manually assigned to each vulnerability on the device and then a score for the device is computed by aggregating the CVSS score of the vulnerabilities within the device. Finally, the devices are ranked based on their score.

Yang et al. [66] evaluated the framework through numerical analysis. A network model that consists of a router, a database server, a gateway device, a web camera, and a smart phone is used. The gateway device and the database server are connected by the router. The authors assume that they have vulnerability information on each device. It is also assumed that each device's vulnerability data are made available by a vulnerability scanning tool. The attributes of the vulnerabilities are retrieved from the NVD database. Vulnerabilities are ranked through a series of computations.

Allouzi and Khan [67] used numerical analysis to validate the framework. The IoMT security threats environment are modelled as a transition graph that consists of a safe state, 12 threat states, and an attack state based on a Markov chain. The analysis considered 11 vulnerabilities along with their CVSS scores. The transition probability of security threats from the safe state to the attack state is quantified using the Markov chain and transit probability matrix. The main concern with this framework is that the threat states do not take into account the countermeasures that may be in place, and the implication of ignoring this was not discussed. Furthermore, only the attacks, not the vulnerabilities, are quantified.

Rizvi et al. [70] used several hypothetical scenarios to validate their frameworks. The scenarios are based on typical IoT devices and domains, such as smart healthcare and smart homes. The model creates an impact metric for each IoT device by generating numerical scores of TUI and CIA (confidentiality, integrity, and availability scores); as well, TUI–CIA metrics were computed. The TUI is a score of transparency, unlinkability, and intervenability. The TUI score is calculated using particular mathematical equations and represents the consequence of adding points based on each question's responses. The CIA score is given based on confidentiality, integrity, and availability of the CVSS, and the TUI–CIA score is a combination of the CIA and TUI scores, resulting in the final impact score for the device vulnerability. The results prove the efficacy of the framework. If implemented in actual systems, the framework will be able to assist users in understanding the privacy of their devices. Nevertheless, it fails to identify potential data privacy vulnerabilities.

Ge et al. [88] considered an IoT network in a smart home with Wi-Fi, a tablet with ZigBee and Android, and a Wi-Fi-enabled smart TV. For the healthcare monitoring case, WBAN was used with a coordinator device (e.g., PDA) and several sensors put on different parts of a human body. For environment monitoring, a WSN composed of a sink node and 1000 sensors was used. The WSN is assumed to be deployed in an unmanned open field. A sinkhole attack was used as an example to show how an attacker could remotely compromise the IoT devices. The authors calculate and assign scores to devices based on the vulnerabilities. The framework is helpful for choosing which devices along the paths should be secured first and for comparing the effectiveness of various device-level techniques based on the evaluation of various security metrics.

5.4.3. Scenario- and Case-Based

Lee et al. [84] evaluated the efficacy and competitiveness of their game-based strategy using a smart home as the use case. The network contained various IoT devices for the home, such as an in-home display (IHD) for querying about electricity use, a customer EMS for managing energy use in a home, an energy service interface (ESI) for direct interaction with external entities to the user domain, and a smart meter for intelligent electricity usage measuring that also serves as a communication gateway. A security vulnerability corresponding to a DoS attack was quantified in the case study. The quantifiable impact value of each action was computed after game state generation. The authors compared their framework against other conventional methods such as CVSS.

Jiang and Atif [74] validated the framework using a vulnerability analysis case study. The authors ran two performance evaluation experiments using 156,040 vulnerability records from existing repositories as well as data from crawled websites. In addition, non-rated CVSS 3.0 reports were used to create 75,265 instances of CVSS 3.0 corpus data. In addition, CVSS severity scores from various CVSS versions were used. When vulnerabilities are sourced from only NVD, the attack vector classifier achieves 90.36% accuracy. The accuracy increases to 93.68% when data are sourced from both NVD and SecurityFocus entries. Shojaeshafiei et al. [87] analysed the vulnerability measurement of portable devices used in organisations. A lost mobile device policy is used as a case study. Fuzzy inference processes are used for vulnerability quantification.

Validation in Qu and Chan [78] is based on a simple scenario using a shopping cart application on a BLE wireless network. In the scenario, two wireless nodes (a customer shopping cart and a BLE beacon in the store) were modelled. There is also one potentially fraudulent entry node used by the attacker to gain access to the wireless network. This simple scenario demonstrates the special case used by the authors, but more tests that incorporate more credible real-time scenarios are needed to demonstrate its efficacy. Moreover, the network structures examined are a miniature of the mundane client-server system.

5.4.4. Testbed

A testbed for evaluation of the models is another common validation technique used by researchers. Oser et al. [55] used a testbed to validate their framework. However, the test was more on the identification of the devices rather than assessing their vulnerabilities and quantifications. Additionally, the evaluation was conducted using 38 device models, which is very small compared to the actual number of IoT devices in their network. The results shows that the framework predicts perfectly the current and future risks of the IoT devices in the network. However, this ideal prediction could be due to the small data sample. Therefore, the prediction performance must be demonstrated in a large-scale, realistic setting.

Similarly, Akhilesh et al. [71] used a testbed to evaluate the framework in terms of run time and device security. The testbed is composed of a Kali Linux server, a Wi-Fi hotspot, five IoT devices that include smart things (plug, bulb, camera), and a Google Home Mini, with the aim of finding the most common vulnerabilities that appear in them. They also calculated a CVSS score based on the base metrics for each device individually.

Biondi et al. [65] examined the security of a particular model of web camera, the TP-Link Tapo C200, to understand fully what Tapo's vulnerabilities may be when the Tapo C200 continues to perform during its operation, with a focus on device initialisation, device use through TP-Link proprietary systems, and device integration with third-party systems. According to the test results, the web camera is vulnerable to three different types of attacks, including "Motion Oracle" denial of service and video eavesdropping. "Motion Oracle" is a new type of attack [65]. The test shows that anyone with rudimentary knowledge of how to operate non-commercial scanning tools can exploit web cameras.

5.4.5. Scanning

Bugeja et al. [75] used Shodan to carried out vulnerability scanning of smart connected cameras. The vulnerability information collected from Shodan is then cross-referenced with the CVE database for vulnerability assessment. The findings indicate that a considerable number of web cameras are vulnerable to a variety of security and privacy flaws. This approach will only identify publicly disclosed security vulnerabilities that have been included in the CVE database. Other vulnerabilities that have not made it to the CVE database are excluded from the result.

McMahon et al. [72] used the combination of Shodan and Nessus to assess medical device vulnerabilities. Vulnerability scanning was done using Shodan. The IP address of the devices discovered from Shodan were passed on to Nessus for scoring the vulnerabilities. Nessus was configured to detect obsolete operating systems, web vulnerabilities, and to check for default login credentials. No port scans were performed, mainly to avoid possible harm to devices. The results from Nessus were stored in a database for manual evaluation and verification. The assessment resulted in the identification of 1604 devices with a total of 3964 vulnerabilities. Some of the devices had vulnerabilities rated as 'Critical' and some as 'High'. A large portion of devices had vulnerabilities rated as 'Medium' or 'Low'. The study did not actually indicate which devices were vulnerable and this information would have been useful to the vendors of the devices to mitigate the vulnerabilities.

Williams et al. [73] followed an identical approach of scanning for vulnerabilities first and then using Nessus to score the detected vulnerabilities, as in [72]. The authors concentrated on IoT devices, which are ubiquitous and have the potential to compromise consumer privacy if not secured. They used 11 keywords such as "smart tv" as inputs to Shodan and retrieved a large number of records for a variety of consumer devices. The output from Shodan was fed into Nessus for scoring vulnerabilities. The results revealed obsolete versions of a network discovery and communication protocol as the most prevalent sever vulnerability for webcams. The simple network management protocol (SNMP) was the most common sever vulnerability for smart TVs.

6. Future Directions

The persistence of the top ten IoT vulnerabilities listed by OWASP (Open Web Application Security Project) [9] reflects the lack of maturity of IoT security vulnerability assessment frameworks. Several future research opportunities exist in relation to the topics discussed in this paper. In this section, we highlight some of these open problems.

The evolution of the IoT has resulted in large IoT networks containing a large number of heterogeneous IoT devices. IoT devices differ considerably in terms of type, manufacturer, hardware, firmware version, communication protocols, and operating environment. Different IoT flaws can arise as a result of different IoT device capabilities, characteristics, and deployment environments. Existing vulnerability assessment frameworks are primarily focused on a specific class of IoT devices. They are limited in terms of device heterogeneity and scalability. Since an approach designed for a specific class of IoT devices may not be viable for IoT systems with heterogeneous device configurations, there is a need to expand these frameworks or develop new approaches that take IoT device variability and different deployment contexts into account.

Another research direction is the development of an IoT-tailored vulnerability quantification approach. Although CVSS was created for conventional IT systems, it is now widely utilised for quantifying IoT vulnerability. CVSS scores are risk metrics that are calculated based on factors such as impact and exploitability ratings for a CVE. Both impact and exploitability factors commonly reflect the non-IoT device parameters. In addition, vulnerabilities must be individually assessed to generate CVSS scores based on the scorer's understanding of the vulnerability. Given the fundamental differences between IoT and traditional IT systems, the use of default impact and exploitability factors does not provide security professionals with insight into the IoT-specific risks that IoT ecosystem vulnerabilities pose. As a result, more research is required to solve this problem.

An IoT ecosystem involves cloud, mobile application, hardware, and physical device vulnerabilities. As a result, efficiently modelling and assessing multi-level vulnerabilities in IoT-enabled ecosystems remains a challenge. Approaches that consider multi-host and multi-stage vulnerability assessment are required.

Automating IoT vulnerability assessment is another future research direction. Given the widespread use of IoT devices in security-sensitive applications such as critical infrastructures [7], it is important to mitigate the impact of IoT vulnerabilities by proactively assessing the risk they pose and prioritising their resolution. Assessing the vulnerabilities of IoT devices is challenging because it requires a thorough understanding of specific IoT vulnerability vectors, which are defined by various IoT-related factors, such as architecture, resource constraints, and protocols. There is the possibility of using machine learning techniques to address these challenges. However, there is very little work that uses machine learning-based approaches. A fuzzy inference system-based approach, for example, is promising, but it currently requires input from security professionals via a questionnaire to quantify vulnerabilities. As an alternative to cybersecurity professionals, a machine learning approach could be developed. The use of graph theory in determining IoT vulnerability scores is another area that requires further research. This is because the dynamic nature of IoT and diverse network topologies for IoT networks pose significant complexity for modelling the attack graph nodes. An attack tree-based approach is another possibility for further research. However, the challenge is how to ensure accurate quantification, because an attack tree cannot adequately represent the procedures involved in a cyberattack because they are all carried out in a single node (i.e., the root node).

It is critical to have timely information about IoT vulnerabilities in order to implement appropriate risk mitigation measures. This requires an efficient way of gathering vulnerability data from IoT ecosystems. Although vulnerabilities in IoT devices are being discovered at a rapid pace, mechanisms to collect this data efficiently need be developed. It is particularly difficult to gather data on IoT vulnerabilities because there are so many different types of IoT devices, and each one can have different hardware, software, operating systems, applications, and protocols. Compounding the challenge is that rarely do available IoT vulnerability data contain information such as the precise IoT architecture layer that the vulnerability affects, information on the physical requirements of IoT devices, and the protocols in use.

Last but not least, the practical application of current IoT vulnerability assessment frameworks in IoT contexts still needs more research. The frameworks studied in this paper used approaches such as simulation- and scenario-based validations under controlled environments. There is a need to validate these frameworks in actual IoT deployment environments and to study the impacts of heterogeneous devices, varying protocols, specifications, and so forth on vulnerability assessment frameworks.

7. Conclusions

The IoT has the potential to transform many aspects of our modern lives, but suffers from major security flaws. This paper presented a state-of-the-art systematic literature review on IoT vulnerability assessment and quantification frameworks, as well as the application domains and validation methods used to demonstrate the general practicability of existing frameworks. Our findings revealed that IoT device deployment is growing rapidly, but security solutions are not developing at the same rate. Specifically, the research in IoT vulnerability assessment is relatively unexplored, despite mounting indications that hackers are getting increasingly interested in exploiting IoT vulnerabilities. We identified gaps and limitations in existing IoT vulnerability assessment methods and suggested that future research should focus on developing vulnerability assessment frameworks that take the distinct features of the IoT into account in order to efficiently assess IoT vulnerabilities. We have highlighted the benefits of machine learning techniques for automating the IoT vulnerability assessment. The insights into current IoT vulnerability assessment approaches, gaps, and research directions provided in this paper are useful for ongoing efforts to characterise cybersecurity risks and manage IoT vulnerabilities, as well as to develop new and efficient IoT vulnerability assessment methods.

Author Contributions: Conceptualization, S.A.B. and J.A.; methodology, S.A.B.; validation, S.A.B. and J.A.; formal analysis, S.A.B. and J.A.; investigation, S.A.B.; writing—original draft preparation, S.A.B.; writing—review and editing, J.A.; supervision, J.A.; project administration, J.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Davis, B.D.; Mason, J.C.; Anwar, M. Vulnerability studies and security postures of IoT devices: A smart home case study. *IEEE Internet Things J.* **2020**, *7*, 10102–10110. [CrossRef]
2. Abawajy, J.H.; Hassan, M.M. Federated internet of things and cloud computing pervasive patient health monitoring system. *IEEE Commun. Mag.* **2017**, *55*, 48–53. [CrossRef]
3. Ghanavati, S.; Abawajy, J.H.; Izadi, D.; Alelaiwi, A.A. Cloud-assisted IoT-based health status monitoring framework. *Clust. Comput.* **2017**, *20*, 1843–1853. [CrossRef]
4. Chen, L.-B.; Huang, G.-Z.; Huang, X.-R.; Wang, W.-C. A Self-Supervised Learning-Based Intelligent Greenhouse Orchid Growth Inspection System for Precision Agriculture. *IEEE Sens. J.* **2022**, *22*, 24567–24577. [CrossRef]
5. Ghosh, A.; Abawajy, J.; Chowdhury, M. Redefining the construction managerial landscape to facilitate Industry 4.0 implementation: Scientometric mapping of research frontiers. *Constr. Innov.* **2022**. [CrossRef]
6. Hassan, M.M.; Huda, S.; Sharmeen, S.; Abawajy, J.; Fortino, G. An adaptive trust boundary protection for IIoT networks using deep-learning feature-extraction-based semisupervised model. *IEEE Trans. Ind. Inform.* **2020**, *17*, 2860–2870. [CrossRef]
7. Kimani, K.; Oduol, V.; Langat, K. Cyber security challenges for IoT-based smart grid networks. *Int. J. Crit. Infrastruct. Prot.* **2019**, *25*, 36–49. [CrossRef]
8. Vailshery, L.S. IoT Connected Devices Worldwide 2030. 2021. Available online: https://www.statista.com/statistics/802690/worldwide-connecteddevices-by-access-technology (accessed on 10 December 2022).
9. OWASP, T.I.V. Top IoT Vulnerabilities. 2016. Available online: https://www.owasp.org/index.php (accessed on 15 October 2022).
10. Bertino, E.; Islam, N. Botnets and internet of things security. *Computer* **2017**, *50*, 76–79. [CrossRef]
11. Arampatzis, A. Top 10 Vulnerabilities That Make IoT Devices Insecure. Available online: https://venafi.com/blog/top-10-vulnerabilities-make-iot-devices-insecure/ (accessed on 29 December 2022).
12. Anand, P.; Singh, Y.; Selwal, A.; Alazab, M.; Tanwar, S.; Kumar, N. IoT Vulnerability Assessment for Sustainable Computing: Threats, Current Solutions, and Open Challenges. *IEEE Access* **2020**, *8*, 168825–168853. [CrossRef]
13. Antonakakis, M.; April, T.; Bailey, M.; Bernhard, M.; Bursztein, E.; Cochran, J.; Durumeric, Z.; Halderman, J.A.; Invernizzi, L.; Kallitsis, M. Understanding the mirai botnet. In Proceedings of the 26th {USENIX} Security Symposium ({USENIX} Security 17), Vancouver, BC, Canada, 16–18 August 2017; pp. 1093–1110.
14. Kolias, C.; Kambourakis, G.; Stavrou, A.; Voas, J. DDoS IoT: Mirai and other botnets. *Computer* **2017**, *50*, 80–84. [CrossRef]

15. Chen, K.; Zhang, S.; Li, Z.; Zhang, Y.; Deng, Q.; Ray, S.; Jin, Y. Internet-of-Things security and vulnerabilities: Taxonomy, challenges, and practice. *J. Hardw. Syst. Secur.* **2018**, *2*, 97–110. [CrossRef]
16. Mosenia, A.; Jha, N.K. A comprehensive study of security of internet-of-things. *IEEE Trans. Emerg. Top. Comput.* **2016**, *5*, 586–602. [CrossRef]
17. Alaba, F.A.; Othman, M.; Hashem, I.A.T.; Alotaibi, F. Internet of Things security: A survey. *J. Netw. Comput. Appl.* **2017**, *88*, 10–28. [CrossRef]
18. Zarpelão, B.B.; Miani, R.S.; Kawakani, C.T.; de Alvarenga, S.C. A survey of intrusion detection in Internet of Things. *J. Netw. Comput. Appl.* **2017**, *84*, 25–37. [CrossRef]
19. Neshenko, N.; Bou-Harb, E.; Crichigno, J.; Kaddoum, G.; Ghani, N. Demystifying IoT security: An exhaustive survey on IoT vulnerabilities and a first empirical look on internet-scale IoT exploitations. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2702–2733. [CrossRef]
20. Ban, X.; Ding, M.; Liu, S.; Chen, C.; Zhang, J. A Survey on IoT Vulnerability Discovery. In Proceedings of the Network and System Security: 16th International Conference (NSS 2022), Denarau Island, Fiji, 9–12 December 2022; pp. 267–282.
21. Yu, M.; Zhuge, J.; Cao, M.; Shi, Z.; Jiang, L. A survey of security vulnerability analysis, discovery, detection, and mitigation on IoT devices. *Future Internet* **2020**, *12*, 27. [CrossRef]
22. Farooq, U.; Tariq, N.; Asim, M.; Baker, T.; Al-Shamma'a, A. Machine learning and the Internet of Things security: Solutions and open challenges. *J. Parallel Distrib. Comput.* **2022**, *162*, 89–104. [CrossRef]
23. Rana, M.; Mamun, Q.; Islam, R. Lightweight cryptography in IoT networks: A survey. *Future Gener. Comput. Syst.* **2022**, *129*, 77–89. [CrossRef]
24. Rytel, M.; Felkner, A.; Janiszewski, M. Towards a safer internet of things—A survey of IoT vulnerability data sources. *Sensors* **2020**, *20*, 5969. [CrossRef]
25. Allifah, N.M.; Zualkernan, I.A. Ranking security of IoT-based smart home consumer devices. *IEEE Access* **2022**, *10*, 18352–18369. [CrossRef]
26. Feng, X.; Zhu, X.; Han, Q.-L.; Zhou, W.; Wen, S.; Xiang, Y. Detecting vulnerability on IoT device firmware: A survey. *IEEE/CAA J. Autom. Sin.* **2022**, *10*, 25–41. [CrossRef]
27. Srivastava, A.; Gupta, S.; Quamara, M.; Chaudhary, P.; Aski, V.J. Future IoT-enabled threats and vulnerabilities: State of the art, challenges, and future prospects. *Int. J. Commun. Syst.* **2020**, *33*, e4443. [CrossRef]
28. Meneghello, F.; Calore, M.; Zucchetto, D.; Polese, M.; Zanella, A. IoT: Internet of threats? A survey of practical security vulnerabilities in real IoT devices. *IEEE Internet Things J.* **2019**, *6*, 8182–8201. [CrossRef]
29. Costin, A. Security of cctv and video surveillance systems: Threats, vulnerabilities, attacks, and mitigations. In Proceedings of the 6th International Workshop on Trustworthy Embedded Devices, Vienna, Austria, 28 October 2016; pp. 45–54.
30. Nadir, I.; Mahmood, H.; Asadullah, G. A taxonomy of IoT firmware security and principal firmware analysis techniques. *Int. J. Crit. Infrastruct. Prot.* **2022**, *38*, 100552. [CrossRef]
31. Wright, C.; Moeglein, W.A.; Bagchi, S.; Kulkarni, M.; Clements, A.A. Challenges in firmware re-hosting, emulation, and analysis. *ACM Comput. Surv. CSUR* **2021**, *54*, 5. [CrossRef]
32. Xie, W.; Jiang, Y.; Tang, Y.; Ding, N.; Gao, Y. Vulnerability detection in iot firmware: A survey. In Proceedings of the 2017 IEEE 23rd International Conference on Parallel and Distributed Systems (ICPADS), Shenzhen, China, 15–17 December 2017; pp. 769–772.
33. Frustaci, M.; Pace, P.; Aloi, G.; Fortino, G. Evaluating critical security issues of the IoT world: Present and future challenges. *IEEE Internet Things J.* **2017**, *5*, 2483–2495. [CrossRef]
34. Qasem, A.; Shirani, P.; Debbabi, M.; Wang, L.; Lebel, B.; Agba, B.L. Automatic vulnerability detection in embedded devices and firmware: Survey and layered taxonomies. *ACM Comput. Surv. CSUR* **2021**, *54*, 25. [CrossRef]
35. Yaqoob, T.; Abbas, H.; Atiquzzaman, M. Security vulnerabilities, attacks, countermeasures, and regulations of networked medical devices—A review. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 3723–3768. [CrossRef]
36. Suo, H.; Wan, J.; Zou, C.; Liu, J. Security in the internet of things: A review. In Proceedings of the 2012 International Conference on Computer Science and Electronics Engineering, Hangzhou, China, 23–25 March 2012; pp. 648–651.
37. Tewari, A.; Gupta, B.B. Security, privacy and trust of different layers in Internet-of-Things (IoTs) framework. *Future Gener. Comput. Syst.* **2020**, *108*, 909–920. [CrossRef]
38. Abbasi, M.; Plaza-Hernández, M.; Prieto, J.; Corchado, J.M. Security in the Internet of Things Application Layer: Requirements, Threats, and Solutions. *IEEE Access* **2022**, *10*, 97197–97216. [CrossRef]
39. Ray, P.P. A survey on Internet of Things architectures. *J. King Saud Univ.-Comput. Inf. Sci.* **2018**, *30*, 291–319.
40. Abawajy, J.; Darem, A.; Alhashmi, A.A. Feature subset selection for malware detection in smart IoT platforms. *Sensors* **2021**, *21*, 1374. [CrossRef]
41. Mahdin, H.; Abawajy, J. An approach for removing redundant data from RFID data streams. *Sensors* **2011**, *11*, 9863–9877. [CrossRef]
42. Deogirikar, J.; Vidhate, A. Security attacks in IoT: A survey. In Proceedings of the 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 10–11 February 2017; pp. 32–37.
43. Varadharajan, V.; Tupakula, U.; Karmakar, K. Study of Security Attacks against IoT Infrastructures. Available online: https://www.newcastle.edu.au/__data/assets/pdf_file/0020/552017/TR1-ISIF-ASIA.pdf (accessed on 7 November 2022).

44. Eresheim, S.; Luh, R.; Schrittwieser, S. On the impact of kernel code vulnerabilities in iot devices. In Proceedings of the 2017 International Conference on Software Security and Assurance (ICSSA), Altoona, PA, USA, 24–25 July 2017; pp. 1–5.
45. Anand, P.; Singh, Y.; Selwal, A.; Singh, P.K.; Felseghi, R.A.; Raboaca, M.S. Iovt: Internet of vulnerable things? Threat architecture, attack surfaces, and vulnerabilities in internet of things and its applications towards smart grids. *Energies* **2020**, *13*, 4813. [CrossRef]
46. Ahmad, I.; Niazy, M.S.; Ziar, R.A.; Khan, S. Survey on IoT: Security threats and applications. *J. Robot. Control* **2021**, *2*, 42–46. [CrossRef]
47. Deloitte. Internet of Things (Iot)—The Rise of the Connected World. Available online: https://www2.deloitte.com/content/dam/Deloitte/in/Documents/technology-media-telecommunications/in-tmt-IoT_Theriseoftheconnectedworld-28aug-noexp.pdf (accessed on 10 December 2022).
48. Symantec, T.H.T. Threat Landscape Trends—Q1 2020. Available online: https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/threat-landscape-q1--2020 (accessed on 15 October 2022).
49. Nebbione, G.; Calzarossa, M.C. Security of IoT application layer protocols: Challenges and findings. *Future Internet* **2020**, *12*, 55. [CrossRef]
50. Tripathi, N.; Hubballi, N. Application layer denial-of-service attacks and defense mechanisms: A survey. *ACM Comput. Surv. CSUR* **2021**, *54*, 86. [CrossRef]
51. Altaf, I.; ul Rashid, F.; Dar, J.A.; Rafiq, M. Vulnerability assessment and patching management. In Proceedings of the 2015 International Conference on Soft Computing Techniques and Implementations (ICSCTI), Faridabad, India, 8–10 October 2015; pp. 16–21.
52. Tenable. Vulnerability Assessment Solution Nessus Professional TM. Available online: https://www.tenable.com/products/nessus/nessus-professional (accessed on 5 October 2022).
53. Matherly, J. *Complete Guide to Shodan: Collect. Analyze. Visualize. Make Internet Intelligence Work for You*; Leanpub: Victoria, BC, Canada, 2016; pp. 1–86.
54. Org, F. Common Vulnerability Scoring System Version 3.1, Specification Document. TLP:WHITE. Available online: https://www.first.org/cvss/v3--1/cvss-v31-specification_r1.pdf (accessed on 20 July 2022).
55. Oser, P.; van der Heijden, R.W.; Lüders, S.; Kargl, F. Risk prediction of IoT devices based on vulnerability analysis. *ACM Trans. Priv. Secur.* **2022**, *25*, 14. [CrossRef]
56. Martin, R.A. *Common Weakness Enumeration*; Mitre Corporation: McLean, VA, USA, 2007; p. 24.
57. Booth, H.; Rike, D.; Witte, G.A. *The National Vulnerability Database (NVD): Overview*; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2013. Available online: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=915172 (accessed on 21 February 2023).
58. Kitchenham, B.; Brereton, P. A systematic review of systematic review process research in software engineering. *Inf. Softw. Technol.* **2013**, *55*, 2049–2075. [CrossRef]
59. Tange, K.; de Donno, M.; Fafoutis, X.; Dragoni, N. A systematic survey of industrial Internet of Things security: Requirements and fog computing opportunities. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2489–2520. [CrossRef]
60. Moher, D.; Liberati, A.; Tetzlaff, J.; Altman, D.G. Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement. *Int. J. Surg.* **2010**, *8*, 336–341. [CrossRef] [PubMed]
61. Radack, S.M. Conducting Security-Related Risk Assessments: Updated Guidelines for Comprehensive Risk Management Programs, ITL Bulletin, National Institute of Standards and Technology, Gaithersburg, MD. Available online: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=912722 (accessed on 27 October 2022).
62. Peters, M.D.; Marnie, C.; Tricco, A.C.; Pollock, D.; Munn, Z.; Alexander, L.; McInerney, P.; Godfrey, C.M.; Khalil, H. Updated methodological guidance for the conduct of scoping reviews. *JBI Evid. Synth.* **2020**, *18*, 2119–2126. [CrossRef] [PubMed]
63. Stellios, I.; Kotzanikolaou, P.; Grigoriadis, C. Assessing IoT enabled cyber-physical attack paths against critical systems. *Comput. Secur.* **2021**, *107*, 102316. [CrossRef]
64. Rizvi, S.; McIntyre, N.; Ryoo, J. Computing security scores for IoT device vulnerabilities. In Proceedings of the 2019 International Conference on Software Security and Assurance (ICSSA), St. Pölten, Austria, 25–26 July 2019; pp. 52–59.
65. Eceiza, M.; Flores, J.L.; Iturbe, M. Fuzzing the internet of things: A review on the techniques and challenges for efficient vulnerability discovery in embedded systems. *IEEE Internet Things J.* **2021**, *8*, 10390–10411. [CrossRef]
66. Biondi, P.; Bognanni, S.; Bella, G. Vulnerability Assessment and Penetration Testing on IP camera. In Proceedings of the 8th International Conference on Internet of Things: Systems, Management and Security (IOTSMS 2021), Gandia, Spain, 6–9 December 2021.
67. Yang, J.; Xue, Y.; Lei, M.; Che, B. Associated Hazard Assessment of IoT Vulnerability Based on Risk Matrix. In Proceedings of the Artificial Intelligence and Security: 6th International Conference (ICAIS 2020), Hohhot, China, 17–20 July 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 525–535.
68. Allouzi, M.A.; Khan, J.I. Identifying and modeling security threats for IoMT edge network using Markov chain and common vulnerability scoring system (CVSS). *arXiv* **2021**, arXiv:2104.11580.
69. Ando, E.; Kayashima, M.; Komoda, N. A Proposal of security requirements definition methodology in connected car systems by CVSS V3. In Proceedings of the 2016 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI), Kumamoto, Japan, 10–14 July 2016; pp. 894–899.

70. Yiğit, B.; Gür, G.; Alagöz, F.; Tellenbach, B. Cost-aware securing of IoT systems using attack graphs. *Ad Hoc Netw.* **2019**, *86*, 23–35. [CrossRef]
71. Rizvi, S.; Williams, I.; Campbell, S. TUI Model for data privacy assessment in IoT networks. *Internet Things* **2022**, *17*, 100465. [CrossRef]
72. Akhilesh, R.; Bills, O.; Chilamkurti, N.; Chowdhury, M.J.M. Automated Penetration Testing Framework for Smart-Home-Based IoT Devices. *Future Internet* **2022**, *14*, 276. [CrossRef]
73. McMahon, E.; Williams, R.; El, M.; Samtani, S.; Patton, M.; Chen, H. Assessing medical device vulnerabilities on the Internet of Things. In Proceedings of the 2017 IEEE International Conference on Intelligence and Security Informatics (ISI), Beijing, China, 22–24 July 2017; pp. 176–178.
74. Williams, R.; McMahon, E.; Samtani, S.; Patton, M.; Chen, H. Identifying vulnerabilities of consumer Internet of Things (IoT) devices: A scalable approach. In Proceedings of the 2017 IEEE International Conference on Intelligence and Security Informatics (ISI), Beijing, China, 22–24 July 2017; pp. 179–181.
75. Jiang, Y.; Atif, Y. Towards automatic discovery and assessment of vulnerability severity in cyber–physical systems. *Array* **2022**, *15*, 100209. [CrossRef]
76. Bugeja, J.; Jönsson, D.; Jacobsson, A. An investigation of vulnerabilities in smart connected cameras. In Proceedings of the 2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), Athens, Greece, 19–23 March 2018; pp. 537–542.
77. Duan, X.; Ge, M.; Le, T.H.M.; Ullah, F.; Gao, S.; Lu, X.; Babar, M.A. Automated security assessment for the internet of things. In Proceedings of the 2021 IEEE 26th Pacific Rim International Symposium on Dependable Computing (PRDC), Perth, Australia, 1–4 December 2021; pp. 47–56.
78. Ur-Rehman, A.; Gondal, I.; Kamruzzaman, J.; Jolfaei, A. Vulnerability modelling for hybrid industrial control system networks. *J. Grid Comput.* **2020**, *18*, 863–878. [CrossRef]
79. Qu, Y.; Chan, P. Assessing vulnerabilities in Bluetooth low energy (BLE) wireless network based IoT systems. In Proceedings of the 2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), New York, NY, USA, 9–10 April 2016; pp. 42–48.
80. Rashed, M.; Kamruzzaman, J.; Gondal, I.; Islam, S. Vulnerability Assessment framework for a Smart Grid. In Proceedings of the 2022 4th Global Power, Energy and Communication Conference (GPECOM), Cappadocia, Turkey, 14–17 June 2022; pp. 449–454.
81. Yadav, G.; Paul, K.; Allakany, A.; Okamura, K. IoT-PEN: An E2E penetration testing framework for IoT. *J. Inf. Process.* **2020**, *28*, 633–642. [CrossRef]
82. George, G.; Thampi, S.M. A graph-based security framework for securing industrial IoT networks from vulnerability exploitations. *IEEE Access* **2018**, *6*, 43586–43601. [CrossRef]
83. Wang, H.; Chen, Z.; Zhao, J.; Di, X.; Liu, D. A vulnerability assessment method in industrial internet of things based on attack graph and maximum flow. *IEEE Access* **2018**, *6*, 8599–8609. [CrossRef]
84. George, G.; Thampi, S.M. Vulnerability-based risk assessment and mitigation strategies for edge devices in the Internet of Things. *Pervasive Mob. Comput.* **2019**, *59*, 101068. [CrossRef]
85. Lee, S.; Kim, S.; Choi, K.; Shon, T. Game theory-based security vulnerability quantification for social internet of things. *Future Gener. Comput. Syst.* **2018**, *82*, 752–760. [CrossRef]
86. Payne, J.; Budhraja, K.; Kundu, A. How secure is your iot network? In Proceedings of the 2019 IEEE International Congress on Internet of Things (ICIOT), Milan, Italy, 8–13 July 2019; pp. 181–188.
87. Ntafloukas, K.; McCrum, D.P.; Pasquale, L. A Cyber-Physical Risk Assessment Approach for Internet of Things Enabled Transportation Infrastructure. *Appl. Sci.* **2022**, *12*, 9241. [CrossRef]
88. Shojaeshafiei, M.; Etzkorn, L.; Anderson, M. Multiple layers of fuzzy logic to quantify vulnerabilies in IoT. *arXiv* **2020**, arXiv:2007.07155.
89. Ge, M.; Hong, J.B.; Guttmann, W.; Kim, D.S. A framework for automating security analysis of the Internet of Things. *J. Netw. Comput. Appl.* **2017**, *83*, 12–27. [CrossRef]
90. Ray, B.R.; Abawajy, J.; Chowdhury, M. Scalable RFID security framework and protocol supporting Internet of Things. *Comput. Netw.* **2014**, *67*, 89–103. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.