information

Review

Monitoring Real Time Security Attacks for IoT Systems Using DevSecOps: A Systematic Literature Review

Ahmed Bahaa [1,2], Ahmed Abdelaziz [3,4,*], Abdalla Sayed [1], Laila Elfangary [1] and Hanan Fahmy [1]

[1] Department of Information Systems, Faculty of Computers and Artificial Intelligence, Helwan University, Helwan 11795, Egypt; ahmed.bahaa@fci.helwan.edu.eg or ahmed.bahaa@fcis.bsu.edu.eg or ahmed.bahaa.farid@gmail.com (A.B.); abdallasayedyessin@gmail.com or abdallasayed@fci.helwan.edu.eg (A.S.); lailaelfangary@gmail.com (L.E.); hanan.fahmy@fci.helwan.edu.eg (H.F.)
[2] Department of Information Systems, Faculty of Computers and Artificial Intelligence, Beni-Suef University, Beni-Suef 62521, Egypt
[3] Information System Department, Nova Information Management School, Universdade Nova de Lisbon, 1099-085 Lisbon, Portugal
[4] Higher Technological Institute, Cairo 11511, Egypt
* Correspondence: D20190535@novaims.unl.pt; Tel.: +351-939-035-914

Abstract: In many enterprises and the private sector, the Internet of Things (IoT) has spread globally. The growing number of different devices connected to the IoT and their various protocols have contributed to the increasing number of attacks, such as denial-of-service (DoS) and remote-to-local (R2L) ones. There are several approaches and techniques that can be used to construct attack detection models, such as machine learning, data mining, and statistical analysis. Nowadays, this technique is commonly used because it can provide precise analysis and results. Therefore, we decided to study the previous literature on the detection of IoT attacks and machine learning in order to understand the process of creating detection models. We also evaluated various datasets used for the models, IoT attack types, independent variables used for the models, evaluation metrics for assessment of models, and monitoring infrastructure using DevSecOps pipelines. We found 49 primary studies, and the detection models were developed using seven different types of machine learning techniques. Most primary studies used IoT device testbed datasets, and others used public datasets such as NSL-KDD and UNSW-NB15. When it comes to measuring the efficiency of models, both numerical and graphical measures are commonly used. Most IoT attacks occur at the network layer according to the literature. If the detection models applied DevSecOps pipelines in development processes for IoT devices, they were more secure. From the results of this paper, we found that machine learning techniques can detect IoT attacks, but there are a few issues in the design of detection models. We also recommend the continued use of hybrid frameworks for the improved detection of IoT attacks, advanced monitoring infrastructure configurations using methods based on software pipelines, and the use of machine learning techniques for advanced supervision and monitoring.

Keywords: DevSecOps; IoT attacks; machine learning; literature review

Citation: Bahaa, A.; Abdelaziz, A.; Sayed, A.; Elfangary, L.; Fahmy, H. Monitoring Real Time Security Attacks for IoT Systems Using DevSecOps: A Systematic Literature Review. *Information* **2021**, *12*, 154. https://doi.org/10.3390/info12040154

Academic Editor: Enrico Denti

Received: 13 March 2021
Accepted: 31 March 2021
Published: 7 April 2021

1. Introduction

In 1999, Kevin Ashton used the term Internet of Things ("IoT") for the first time in the supply chain management context, but it is now used from a general perspective [1]. The Internet of Things (IoT) includes infrastructures of systems, people, interconnected entities, and information resources integrated with services that manipulate information [2]. IoT systems are distributed dynamically and incorporate edge, cloud, and fog computing methods based on the allocation of information and computational resources [3]. IoT devices should cooperate with each other [4]. IoT devices communicate with each other through wireless communication systems and transfer information to a centralized system [5].

According to Statistica [6], there will be approximately 75.44 billion connected IoT devices by 2025 (see Figure 1) [7]. The high amount of IoT devices can pose a major security risk—e.g., malicious software can lead to distributed denial-of-service (DDoS) attacks that target information systems or websites. Mirai malware was used on 21 October 2016 to attack many IoT devices and conducted a major DDoS attack [8,9]. Statistics indicate that 70% of the devices on the IoT are easy to attack [10,11]. For this reason, IoT attack detection is a very important research area. A report released by MacAfee in 2020 showed that cybercriminals are taking advantage of the COVID-19 pandemic, leading to several increased threats, such as PowerShell malware, IoT malware, and mobile malware. In the first quarter of 2020, McAfee labs perceived 419 cyberthreats per minute [12]. Blockchain technology has been used widely in a wide range of applications. Blockchain technology needs a decentralized data management system for sharing and storing the transactions and data in the network. Many of the problems with cyber-physical systems in the IoT systems can be solved by using blockchain technology. Moreover, blockchains help different privacy-preserving models for IoT systems, such as user privacy, data privacy, privacy-preserving aggregation, and location privacy [13].

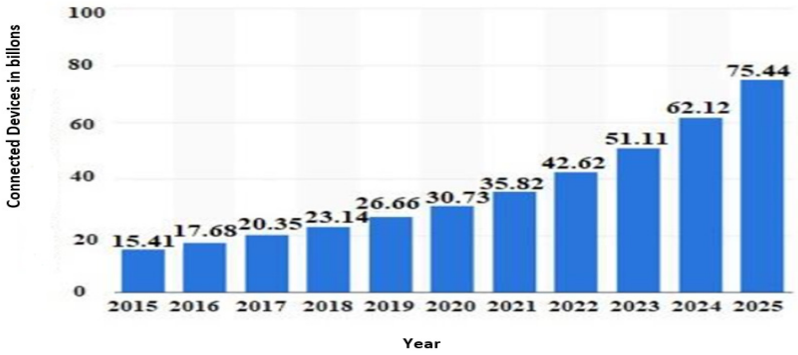

Figure 1. Number of Internet of Things (IoT) connected devices estimated up until 2025 [7].

In addition, IoT devices produce an enormous amount of data [14]. The main method of dealing with big data today is machine learning [15]. Machine learning pipelines that conduct feature extraction, data collection, and binary classification for IoT traffic detection have been developed for many models or systems. Various machine learning algorithms are used for IoT attack detection, such as Bayesian networks (BNs), decision trees (DTs), neural networks (NNs), clustering, support vector machines (SVMs), ensemble learning (EL), and feature selection (FS). Different IoT attacks have also been detected by such proposed models or systems, such as denial-of-service (DoS), remote-to-local (R2L), user-to-root (U2R), and probing attacks. Different datasets are publicly accessible to researchers to use in intrusion detection systems in the IoT, such as KDDCUP99, NSLKDD, and UNSW-NB15. In order to verify the efficiency of these proposed models, various types of evaluation metrics are used for assessment, such as accuracy, recall, and precision. Few studies have analyzed device log traces from IoT devices to identify IoT attacks and monitor infrastructure using DevSecOps.

Our study concentrates on different areas in the detection of IoT attacks. The aim of this study is to analyze, summarize, and evaluate the machine learning techniques used in the detection of IoT attacks. Moreover, we evaluate various datasets used for the models, IoT attack types, independent variables used for the models, evaluation metrics for the assessment of models, and monitoring infrastructures using DevSecOps pipelines. We recommend necessary methods and techniques for upcoming studies.

Darko et al. [16] introduced all studies that used machine learning methods and techniques to enhance IoT security. The authors identified challenges and ideas for future research for the enhancement of IoT security. Sanaz et al. [17] performed a systematic literature review (SLR) of different authentication mechanisms for IoT system security. The authors reviewed various ways to implement authentication in IoT perimeters to identify

recommendations for future studies. Francesca et al. [18] surveyed the security risks in IoT systems and discussed counteractions. Aly et al. [19] performed an SLR and analyzed the security issues related to IoT based upon various layers. Luqman et al. [20] performed an SLR based on the privacy of the IoT system. The authors identified challenges with regard to the privacy of the IoT system exposed, type of attacks occur in the IoT system and recommendations for future studies. Ihsan et al. [20] performed an SLR based on IoT-based botnet attacks. The authors evaluated evaluation metrics for assessment of models, various datasets used for the models, and network forensic methods. Most of the proposed systematic literature reviews (SLRs) focused on authentication mechanisms, privacy, botnet attack avoidance or detection, security risks, and security aspects, while this study aims to (1) analyze, summarize, and evaluate the techniques of machine learning for analyzing device log trace from IoT devices to identify IoT attacks using DevSecOps pipelines and (2) monitor the infrastructure that is created and configured automatically.

The rest of this paper is organized as follows: Section 2 discusses the research methodology. Section 3 describes and analyzes the selected primary studies. The last section concludes the paper and provides recommendations for upcoming work.

2. Research Methodology

The systematic literature review (SLR) methodology was selected to study IoT attack detection models. An SLR involves understanding, evaluating, and identifying the available research evidence to answer specified review questions [21].

2.1. Review Questions (RQs)

For the assessment and the reviewing of primary studies, research questions are listed here. Population, Intervention, Comparison, Outcomes, and Context (PICOC) criteria were used to design these questions [22]. Table 1 illustrates the population, intervention, comparison, outcomes, and context (PICOC) criteria. In this study, the research questions that will be answered are as follows:

Table 1. Population, Intervention, Comparison, Outcomes, and Context (PICOC) criteria.

Population	**IoT Attack Detection**
Intervention	Machine learning techniques
Comparison	Not available
Outcomes	Monitoring real-time security attacks for IoT systems using DevSecOps pipelines
Context	Review the existing studies monitoring real-time security attacks for IoT systems

RQ1—Which datasets have been used for IoT attack detection?

RQ2—What machine learning techniques have been used to detect IoT attacks?

RQ3—What are the current kinds of IoT system attacks that will be detected using machine learning techniques?

RQ4—What are the dependent or independent variables considered when IoT attacks are detected?

RQ5—Which evaluation metrics have been used to evaluate IoT attack detection models?

RQ6—Are the existing models monitoring real-time security for IoT systems using DevSecOps?

2.2. Review Protocol

The process of our study search consisted of selecting digital repositories, creating a search string, proceeding with an initial search, and fetching the first collection of primary studies from digital repositories. We used five digital libraries that have been used in many SLRs related to software engineering [22]: Springer Link, Science Direct, Association for Computer Machinery (ACM), Scopus, and IEEE Xplore. After selecting the repositories, a

search string was needed to perform an exhaustive search to select the main studies. The four steps for defining the search string were [22]:

1. using research questions to define major terms through recognizing population, context, intervention, and outcome;
2. identifying synonyms and alternative spellings for each major term;
3. verifying the search terms in titles, abstracts, and keywords;
4. utilizing the Boolean conjunction operator and/or when producing a search string.

We used the following search string using the steps described above: (IoT OR "Internet of things") AND ("attacks") AND ("Real Time monitor*" OR "Cybersecurity" OR "At-tack detect*" OR ("Intrusion detection*")) AND ("machine learning" OR "supervised learning" OR "classification" OR "regression" OR "unsupervised learning") OR ("DevSecOps").

We used these search strings to collect all available papers in the digital libraries mentioned above. In order to gather as much of the applicable literature as possible, no date limit was placed on the search process in this study. In order to choose the primary studies from the initial list, inclusion and exclusion criteria were designed.

Inclusion criteria:

- written in English;
- related to IoT attack detection;
- published in a journal or conference;
- peer-reviewed papers.

Exclusion criteria:

- focused on detection methods other than machine learning;
- without empirical analysis or results;
- without surveys;
- the full text is not available.

We collected a total of 2898 initial studies from five digital repositories based on the search string. We eliminated primary studies based on the title, abstract, and keywords, which led us to 423 primary studies. The primary studies were carefully reviewed by applying the exclusion and inclusion criteria and finally were reduced to 49 studies. Table 2 illustrates the data sources and search results.

Table 2. Summary of data sources and search results.

Resource Name	Total Results	Initial Selection	Final Selection
IEEE Xplore	248	90	35
Science Direct	746	84	3
Springer Link	1200	150	4
ACM	475	53	2
Scopus	229	46	5
Total	2898	423	49

2.3. Data Extraction

The primary studies used to collect data and answer the research questions in this study were taken from digital repositories. Table 3 shows the characteristics used to answer the questions. Table 4 below summarizes the primary studies that used IoT device testbed datasets with information on machine learning (ML) techniques, IoT attacks, evaluation metrics, and monitoring real-time security using DevSecOps. IoT device testbed datasets were generated from various IoT devices with real traffic. Tables 5–7 below summarize the primary studies using the NSL-KDD, KDDCUP99 or UNSW-NB15 datasets with information on ML techniques, IoT attacks, evaluation metrics, monitoring of real-time

security using DevSecOps, and other datasets used in the primary studies. The KDD-CUP99, NSLKDD, and UNSW-NB15 datasets have been generated for evaluating intrusion detection systems (IDSs). Table 8 below summarizes the primary studies using other public datasets on ML techniques, IoT attacks, evaluation metrics, monitoring of real-time security using DevSecOps, and datasets used in the primary studies. Most of the primary studies used seven different types of machine learning techniques, such as NN, BN, DT, SVM, clustering, FS, and EL. The NN technique has been widely used to enhance the representation of data to build better models. The BN technique manage features separately and thus cannot collect useful information from relations and coordination between features. The DT technique is a popular classification technique for machine learning based on the strategy of divide and conquer. The SVM technique is a supervised learning approach utilized for regression and classification. The clustering technique is suitable when no class of attacks is present. K-nearest neighbors and K-means are two of the clustering algorithms. The FS technique is used to reduce the dimension of data and enhance the technique's performance. EL aims to enhance the results of classification by integrating several models.

Table 3. Data extraction characteristics related to the research questions.

Characteristic	Research Question
Authors, study title, publication year, publication title, source, source type	General
IoT attack datasets	RQ1
IoT attack detection machine learning techniques	RQ2
Type of IoT attacks	RQ3
Dependent/independent variables	RQ4
Performance measures	RQ5
Monitoring real-time security for IoT systems using DevSecOps pipelines	RQ6

Table 4. Internet of Things (IoT) device testbed datasets.

S.No	Reference	ML Techniques	IoT Attacks	Evaluation Metrics	Monitoring Real-Time Security Using DevSecOps
S1	(Eirini et al., 2018) [23]	BN.	Probing, and DOS	Precision, recall, and F-measure.	No.
S3	(Eirini et al., 2019) [24]	BN and DT.	DOS, MITM, reconnaissance, and replay.	Precision, recall, and F-measure.	No.
S6	(Prachi et al., 2017) [25]	DT and Clustering.	Wormhole	Detection rate.	No.
S7	(Fariz et al., 2019) [26]	DT	DOS	Accuracy.	No.
S8	(Aymen et al., 2019) [27]	SVM	Selective forwarding attack.	Accuracy.	No.
S10	(Maede et al., 2019) [28]	DT, SVM and NN.	Backdoor, command injection, and SQL injection.	Accuracy, false alarm rate, ROC curve, and sensitivity matric.	No.
S17	(Parth et al., 2018) [29]	NN.	DOS.	Accuracy, true positive rate, and false positive rate.	No.
S18	(Christiana et al., 2019) [30]	SVM.	Selective forwarding, and blackhole.	Accuracy.	No.

Table 4. *Cont.*

S.No	Reference	ML Techniques	IoT Attacks	Evaluation Metrics	Monitoring Real-Time Security Using DevSecOps
S20	(Suman et al., 2017) [31]	SVM.	DOS.	Precision and recall.	No.
S21	(Mehdi et al., 2016) [32]	SVM.	DOS and DDoS.	Precision and recall.	No.
S22	(Jessica et al., 2019) [33]	NN.	DOS	Detection rate.	Yes.
S24	(Randeep et al., 2019) [34]	NN.	DOS	Precision, recall, and F-score.	No.
S26	(Nadia et al., 2019) [35]	BN, DT and NN.	Amplification	Recall, accuracy, precision, and false positive rate.	No.
S30	(Naoki et al., 2018) [36]	NN.	Wormhole	Detection rate.	No.
S31	(Seiichi et al., 2019) [37]	NN.	Wormhole	Detection rate.	No.
S32	(Yassine, 2019) [38]	DT, Clustering and NN.	Wormhole	Accuracy, precision, and recall.	No.
S33	(Geethapriya et al., 2019) [39]	NN.	Wormhole	Precision, recall, and F1 score.	No.
S38	(Christiana et al., 2020) [40]	SVM.	Blackhole and selective forwarding.	Accuracy, precision, negative predictive value (NPV), recall, and the Matthews correlation coefficient (MCC).	No.
S41	(Zhipeng et al., 2020) [41]	SVM, clustering and DT.	DOS, scanning and MITM.	Accuracy, recall, and F1 score	No.
S47	(Riccardo et al., 2020) [42]	BN.	DOS, scanning and MITM	Accuracy, precision, recall, and F-measure	No.

Table 5. NSL-KDD dataset.

S.No	Reference	ML Techniques	IoT Attacks	Evaluation Metrics	Monitoring Real-Time Security Using DevSecOps	Other Datasets Used
S5	(Chao et al., 2019) [43]	NN.	DOS, U2R and R2L.	Accuracy, precision, and recall.	No.	-
S9	(Poulmanogo et al., 2019) [44]	DT and NB.	Probing, U2R and R2L.	Accuracy.	No.	KDDCUP99.
S12	(Hamed et al., 2019) [45]	Clustering and NB.	U2R and R2L.	Detection rate, and false alarm rate.	No.	-
S13	(Abebe et al., 2018) [46]	NN.	Probing, U2R and R2L.	Accuracy, detection rate, false alarm rate and ROC curve.	No.	-

Table 5. *Cont.*

S.No	Reference	ML Techniques	IoT Attacks	Evaluation Metrics	Monitoring Real-Time Security Using DevSecOps	Other Datasets Used
S25	(Andrii et al., 2019) [47]	NN.	Probing, DOS, U2R and R2L.	Detection rate.	No.	-
S27	(Shahadate et al., 2019) [48]	NN.	Probing.	Accuracy.	No.	-
S28	(Shailendra et al., 2018) [49].	Clustering.	Probing, DOS, U2R and R2L.	Accuracy, sensitivity, F-score and positive predictive value	No.	-
S29	(Abebe et al., 2018) [50]	NN.	Probing, U2R and R2L.	Accuracy, detection rate, false alarm rate, precision, and recall.	No.	-
S34	(Samir et al., 2019) [51]	BN, DT and Clustering.	DOS, Reconnaissance U2R, R2L., Backdoor, Analysis, generic, fuzzers, and shellcode.	Accuracy, false positive rate, precision, and F1-Score.	No.	UNSW-NB15 and KDDCUP99.
S35	(Abhishek et al., 2019) [52]	DT.	DOS	Accuracy, specificity, sensitivity and false positive rate.	No.	CICIDS2017 and UNSW-NB15.
S37	(AHMED et al., 2020) [53]	NN.	DOS and SQL injection.	Accuracy, precision and recall.	No.	UNSW-NB15, CICIDS2017, RPL-NIDDS17 and BoT-IoT
S39	(Seyedeh et al., 2020) [54]	SVM, DT, Clustering.	DOS, U2R and R2L.	Accuracy, precision, recall, and F1-score	No.	-
S44	(Sara et al., 2020) [55]	NN.	U2R and R2L.	Accuracy, F1-score, precision, and recall.	No.	-
S45	(Cristiano et al., 2020) [56]	NN and clustering.	DOS.	Accuracy, Fl-score, precision, and recall	No.	CICIDS2017
S46	(Deepa et al., 2020) [57]	DT	Probing, DOS, U2R and R2L.	Accuracy, Fl-score, precision and recall.	No.	KDDCUP99.

Table 6. KDDCUP99 dataset.

S.No	Reference	ML Techniques	IoT Attacks	Evaluation Metrics	Monitoring Real-Time Security Using DevSecOps	Other Datasets Used
S4	(Shengchu et al., 2017) [58]	Clustering.	Probing, DOS, U2R, and R2L.	Detection rate, false alarm rate and Accuracy	No.	-
S11	(Ionut et al., 2016) [59]	DT, SVM and Clustering.	Probing, DOS, U2R, and R2L.	Precision.	No.	-
S48	(Shubhra et al., 2020) [60]	SVM and BN.	DDoS	Accuracy, sensitivity, specificity, precision, f-measure, AUC (Area under curve) and false positive rate	No.	CAIDA, CONFICKER Worm, and UNINA traffic traces.

Table 7. UNSW-NB15 dataset.

S.No	Reference	ML Techniques	IoT Attacks	Evaluation Metrics	Monitoring Real-Time Security Using DevSecOps	Other Datasets Used
S16	(Bipraneel et al., 2018) [61]	NN.	Reconnaissance, DOS, wormhole and backdoor.	Accuracy, precision, recall, F-measure, miscalculation rate, and detection rate.	No.	-
S19	(Sohaib et al., 2019) [62]	NN.	Reconnaissance, DOS, probing, wormhole and backdoor.	Accuracy and precision	No.	-
S36	(Shahid et al., 2020) [63]	NN.	Reconnaissance, DOS, wormhole, exploits and backdoor.	Accuracy.	No.	-
S42	(Zina et al., 2020) [64]	DT.	Reconnaissance, DOS, wormhole and backdoor.	Accuracy	No.	-

Table 8. Other public datasets.

S.No	Reference	ML Techniques	IoT Attacks	Evaluation Metrics	Monitoring Real-Time Security Using DevSecOps	Used Datasets
S2	(Abhishek et al., 2019) [65]	EL	Sinkhole, local repair attacks, blackhole, sybil, DDOS, hello flooding and selective forwarding.	Accuracy and AUC	No.	RPL-NIDDS17
S14	(Vladimir et al., 2019) [66]	SVM, NN and Clustering.	DOS	Accuracy	No.	CICIDS2017
S15	(Mengmeng et al., 2019) [67]	NN.	Information theft attacks, DDOS, reconnaissance and DOS.	Accuracy, precision, recall, and F-measure.	No.	BoT-IoT
S23	(Mostafa et al., 2018) [68]	Clustering.	Probing, DOS, U2R and R2L.	Detection rate, False Positive rate and Accuracy.	No.	intelIoT
S40	(SHAHID et al., 2020) [69]	NN.	Probing and DOS.	Accuracy, precision, recall and F1 score.	No.	DS2OS
S43	(Monika et al., 2020) [70]	NN.	DDOS.	Accuracy, Fl-score, precision, and recall	No.	CICIDS2017
S49	(Haifaa et al., 2020) [71]	NN.	DOS.	F1 score	No.	MedBIoT.

3. Result

3.1. Datasets

A dataset is classified as a collection of information used in a specific domain. Twenty of the primary studies we identified used IoT device testbed datasets, and the others used public datasets, as shown in Figure 2. IoT device testbed datasets were generated from various IoT devices with real traffic, such as Samsung smart things Hub, smart cameras, smartphones, IoT hubs, intelligent thermostat, and smart assistant speakers. Different datasets are publicly accessible for use in intrusion detection systems (IDSs) for IoT systems. However, public datasets have quality issues. Various public network datasets, for example, KDDCUP99, NSLKDD, and UNSW-NB15, have been generated to evaluate IDSs; however, they do not contain any specific characteristics of IoT systems [72]. The NSL-KDD dataset was built from the KDDCUPP99 datasets [73]. The KDDCUP99 dataset contains a large number of duplicate records that were removed in the NSL-KDD dataset [73]. UNSW-NB15 is different from other datasets such as KDDCUPP99, which has fewer features [74]. The KDDCUP99and NSL-KDD datasets do not contain a set of attack types, while the CICIDS2017 dataset contains a new IoT attack generated from real network traffic such as structured query language (SQL) injection, brute force, XSS, Botnet, web attack, and infiltration [75]. The NSL-KDD and KDDCUP99 datasets are not suitable for evaluating network intrusion detection systems (NIDSs) for IOT; however, the RPL-NIDDS17 dataset includes attack and normal network traffic. Due to the different nature of the datasets, many researchers have used various public datasets in the same primary studies.

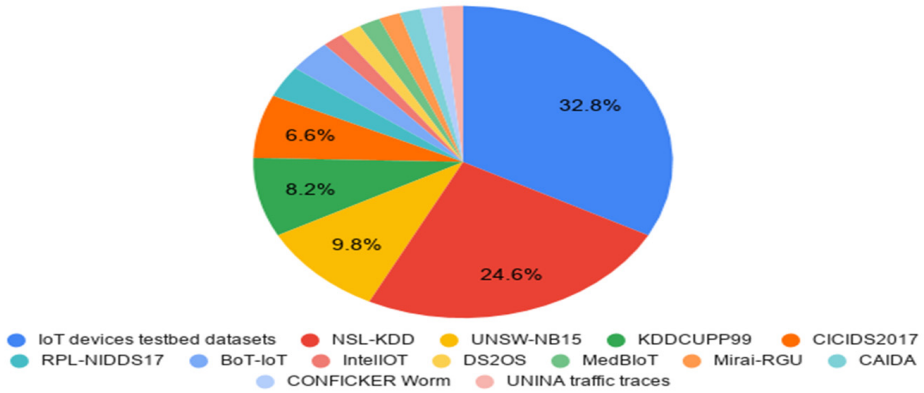

Figure 2. Distribution of IoT attack datasets.

In our study, we observed that the NSL-KDD dataset was used in 15 primary studies. The NSL-KDD dataset was created using three different protocols (TCP, UDP and ICMP). Two test datasets were developed by NSL-KDD—namely, KDDTest+ and KDDTest-21, which have 41 features [76]. This dataset is grouped into different attack categories—namely, R2L, Probe, U2R, and DoS.

The UNSW-NB15 datasets, used in six primary studies, were generated by the Australian Centre's Cyber Range Lab [74]. This dataset varies from previous datasets such as NSL-KDD, which has fewer networks, more repetition, and fewer features. The UNSW-NB15 datasets include 49 features and nine attacks. These attacks include backdoors, fuzzers, analysis, exploits, generic, reconnaissance, shellcode, worms, and DoS.

KDDCUP99, used in five primary studies, was generated by DARPA [73]. This dataset contains around 5 million samples of network captured packets. The KDDCUPP99 datasets have 41 features and three attacks. These attacks include DoS, Probe, R2L, and U2R. The KDDCUP99 datasets contain many redundant and duplicated records.

Other datasets that are rarely used in the primary studies are CICIDS2017, RPL-NIDDS17, BoT-IoT, intelIoT, MedBIoT, CAIDA, CONFICKER Worm, UNINA traffic traces, and DS2OS. CICIDS2017, used in four primary studies, was generated by the Cyber Range Lab of the center of UNSW Canberra Cyber. These datasets have 78 features and eight attacks. These attacks include SQL injection, brute force secure shell protocol (SSH), heartbleed, brute force file transfer protocol (FTP), web attack, DDoS, DoS, botnet, and infiltration, which are not found in other datasets, such as KDDCUP99 and NSL-KDD. RPL-NIDDS17, used in two primary studies, was generated using the NetSim tool. These datasets have 20 features, 2 additional labeling attributes and 7 attacks. These attacks include blackhole, sinkhole, sybil, clone ID, selective forwarding, hello flooding and local repair attacks. BoT-IoT, used in two primary studies, was generated by the Cyber Range Lab of the center of UNSW Canberra Cyber. This dataset contains around 72 million records of network traffic captured from the IoT environment. The BoT-IoT datasets have 32 features and five attacks. These attacks include DoS, DDoS, keylogging, data exfiltration, and service scan. IntelIoT, used in one primary study, was generated by Samuel Madden at the intel research laboratory. This dataset contains around 2 million records captured from 54 sensors spread around the laboratory. For the intelIoT, 30% of all of records became abnormal and the rest of them (70%) became normal. CAIDA, used in one primary study, was generated by the Center for Applied Internet Data Analysis institute. The CAIDA datasets contain unusual traffic traces from DDoS attacks. CONFICKER Worm, used in one primary study, was generated by Center for Applied Internet Data Analysis institute. This dataset was collected from the UCSD Network Telescope after 3 days of network study. DS2OS, used in one primary study, was generated by Kaggle. This dataset contains attacks on sensors and applications; therefore, it consists of 357,952 records, 13 features, and 8 attacks. These attacks include DoS, malicious control, probing, scan, wrong setup, spying,

and normal and malicious operation. The UNINA dataset contains traffic of WAN access router at the University of Napoli Federico.

3.2. Machine Learning Techniques

Many techniques for IoT attack detection have been introduced in the literature, amounting to 49 studies. In this paper we classify primary studies into seven techniques used in IoT attack detection. Most of the primary studies use more than one technique in IoT attack detection. The distribution of the machine learning techniques is shown in Figure 3. The seven techniques presented are BN, DT, NN, clustering, SVM, FS, and EL.

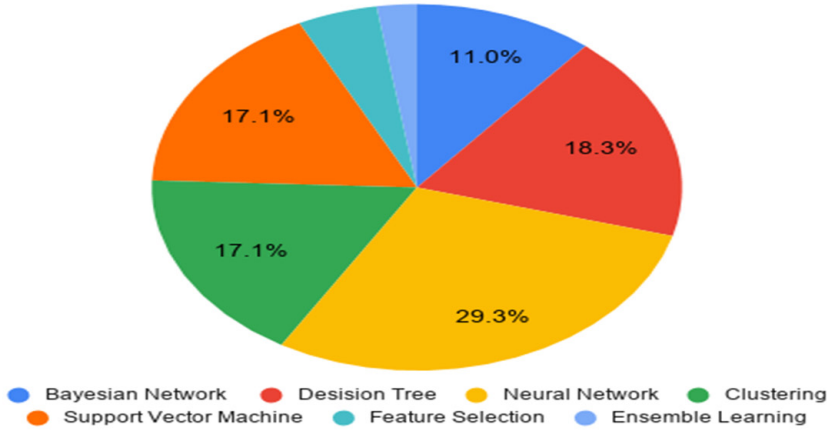

Figure 3. Distribution of machine learning techniques.

NNs are most widely used in IoT attack detection in primary studies. There are many different NN models, such as the convolutional neural network (CNN), deep neural network (DNN), recurrent neural network (RNN), deep learning (DL), and shallow learning. In IDSs, NN techniques have been widely used to enhance the representation of data to build better models. The processing time of NN techniques is high because they have several parameters that need to be tuned, such as the number of neurons in each layer and the number of layers used. Abebe et al. [50] and Abebe et al. [46] proposed a distributed attack detection model based on DL techniques. The proposed model deployed the deep learning model on multiple coordinated nodes for distributed attack detection. Moreover, Ahmed et al. [53] proposed a distributed architecture of an LSTM DL Model deployed on distributed fog nodes, which was managed and modified via a service layer in a cloud computing architecture. This achieved better distributed attack detection than a centralized algorithm. Shahadate et al. [48] also proposed a new model; they combined an autoencoded and dense neural network to detect IoT attacks in the network layer. The autoencoded network provided unsupervised pretraining on the data for less input data noise. A dense neural network was used for final classification in an intrusion detection scenario. The proposed system yielded better results than those acquired when only a DNN was used. There is also a study on combining a CNN and an LSTM by Monika et al. [70], where the aim was to detect IoT attacks. The proposed system achieved good performance and a high detection rate compared to using only MLP, SVM, NB, and random forest. Randeep et al. [34] proposed a model using unsupervised classifiers, such as an autoencoder and PCA, and supervised classifiers, such as the SVM. It was observed that each of these unsupervised machine learning (ML) classifiers performed better than a supervised classifier with new and unseen attacks. Bipraneel et al. [61] proposed a new IDS for detecting IoT attacks based on a bidirectional long short-term memory recurrent neural network (BLSTM RNN). The proposed model learned effectively in the training phase. Shahid et al. [69] proposed a new IDS based on a random neural network (RaNN) approach. The proposed prediction based on the RaNN achieved a higher performance than other machine learning algorithms such as ANN, SVM, and DT. A new IDS using a

DNN algorithm was suggested by Chao et al. [43]. The proposed model achieved a high efficiency for detecting transport layer attacks.

The DT is the second most widely used model in IoT attack detection in primary studies. It is a popular classification technique for machine learning based on the strategy of divide and conquer. It contains nodes and leaves, where the leaves are the class labels and the nodes are one of the features. As a result of its construction, DT requires large storage capacity. Zina et al. [64] proposed a hybrid IDS using random forest (RF), classification and regression tree (CART) algorithms. The RF algorithm was used in feature selection to reduce the dimensions of the dataset into the most significant features. The CART classifier was used to identify different IoT attack classes. Maede et al. [28] proposed an ML-based IDS using seven techniques for the IDS: SVM, KNN, NB, RF, DT, LR, and ANN. RF exhibited the best performance, and NB was the worst in the proposed system. Nadia et al. [35] proposed an IDS at the service layer based on NB, multilayer perceptron (MLP), J48, RF, and sequential minimal optimization (SMO). J48 achieved the best results in binary classification and multiclass classification. NB had the fastest time for CPU training and the worst performance. Yassine et al. [50] proposed an IDS using NB, KNN, RF, SVM, and MLP for detecting IoT attacks. In the proposed IDS, RF achieved the highest performance when detecting routing attacks among all algorithms. Samir et al. [51] proposed a system for the detection of IoT attacks based on NB, LR, DT, RF, KNN, SVM, and MLP algorithms. DT and KNN obtained the best performance among all algorithms; however, compared to the DT algorithm, the KNN needed a high amount of time to classify. Deepa et al. [57] proposed a NIDS based on the RF classifier with a minimal feature set. The proposed system took less time to learn and predict. Fariz et al. [26] proposed middleware using an IDS based on the J48 algorithm to detect DoS attacks. Before using the J48 algorithm, the proposed model cleaned noise from the data.

The SVM is the third most widely used model in IoT attack detection in primary studies. SVM is a supervised learning approach utilized for regression and classification. The SVM maps input vectors into a multidimensional space. They can perform under binary as well as multiclass conditions. For large datasets, SVM is not recommended as the training takes a long time [35,41]. Suman et al. [31] proposed an IDS for IoT security based on SDN strategies which aimed to detect anomalous activity early and enhance resilience. The proposed system was compared with a nonlinear and linear SVM for IoT attack detection. In the proposed IDS, the better learning strategy for identification of attacks was the nonlinear SVM. Christiana et al. [30] proposed a c-SVM machine learning model as an anomaly IDS. The proposed model achieved high detection accuracy when the Sinkhole and Blackhole attacks were present.

One of the unsupervised learning methods is the clustering technique, which is suitable when no class of attacks is present. K-nearest neighbors (KNN) is one of the clustering algorithms. KNN was grouped and trained by certain criteria and analyzed to set in similar K neighbors. Deciding the optimal estimation of K can be a complicated and tedious procedure. Cristiano et al. [56] proposed a hybrid binary classification method based on DNN and KNN. The proposed system gave better results compared to when only DNN or KNN were used. The memory and processing cost worked with low overheads in the proposed method. Shengchu et al. [58] proposed a new model for an IDS, which depends on a dimension reduction algorithm and a classifier. This model used two classifiers: the softmax regression and KNN algorithms. Both algorithms showed equal accuracy, but the softmax regression showed better time efficiency. Mostafa et al. [68] proposed a hybrid model based on K-means and sequential minimal optimization (SMO) for IoT attack detection. K-means clustering was used in the proposed model to cluster the input dataset, and SMO was used to label data whose label was not fixed. The hybrid method gave better results compared to when only SMO or K-means were used.

Bayesian algorithms, specifically naïve Bayes (NB), are the fifth most widely used model in IoT attack detection in primary studies. NB is well known for its simplicity of use, fewer training requirements, and the low time consumption. It manages features

separately and thus cannot collect useful information from relations and coordination between features. Eirini et al. [23] proposed a new model capable of predicting malicious behavior and detecting malicious IoT nodes on a network using NB.

FS is used to reduce the dimension of data and enhance the technique's performance, and some studies have used it to select the best features to be used for IoT attack detection model [43,46,55,60].

The EL techniques are rarely used in IoT attack detection in primary studies. The aim of ELs is to enhance the results of classification by integrating several models. Thus, using many models can increase the accuracy of detection. Abhishek et al. [65] proposed an EL-based network intrusion detection system (ELNIDS), which is based on EL and uses four types of classifiers: Bagged Trees, Boosted Trees, RUSBoosted Trees, and Subspace Discriminant. Boosted Trees and RUSBoosted Trees achieved the best performance in ELNIDS.

3.3. IoT attacks

IoT architecture can be separated into a perception layer, a network layer, a processing layer, and an application layer [77]. There are different features for each IoT layer, so there are multiple threats for each layer [78]. IoT attacks can be detected in any layer of IoT architecture. In the perception layer, hardware components of IoT systems, such as zigbee, radiofrequency identification (RFID), wireless sensor networks (WSNs), and sensors, are vulnerable to various attacks. The network layer in an IoT system has substantial security measures, but certain issues still occur. There are various types of IoT system at-tacks, such as DoS attacks, viruses, man-in-the-middle attacks, and eavesdropping attacks that affect the network layer [79]. The processing layer contains various types of technology, such as data analysis and data storage. The most popular type of attack on the IoT processing layer is a cloud attack since the cloud receives data sent at this phase [80]. The attacker will use trojan worms, horse applications, spyware, malware, and malicious scripting software attacks that can damage IoT system devices in the application layer. Figure 4 illustrates the percentage of IoT attacks considered in primary studies.

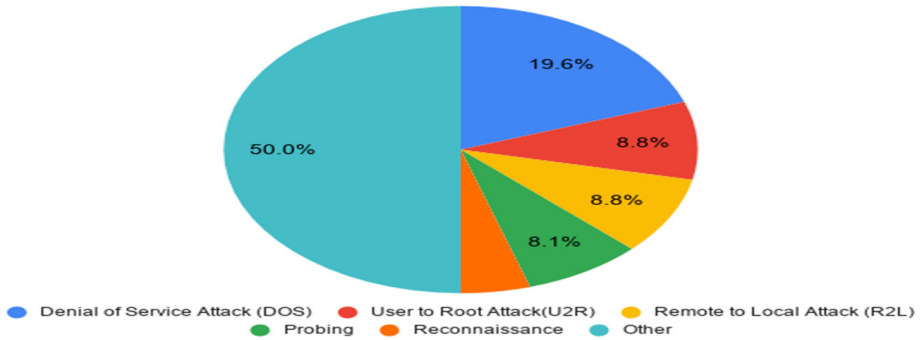

Figure 4. Percentage of IoT attacks considered in primary studies.

DoS attacks were frequently used in the studies we compiled. A DoS attack is a type of attack in which the attacker makes a service inaccessible and stops legal users of the service by sending floods of ICMP echo replies or SYN to port(s). U2R attacks are the second most frequent IoT attack. An U2R attack is when the attacker uses illegal techniques and methods (e.g., sniffing passwords or malicious injection) to gain access to devices or get access from a normal user account. R2L attacks are the third most frequent IoT attack. R2L attacks are exploitations in which the attacker identifies a security vulnerability in a network in order to enter it as a local user. Probing is the fourth most frequent IoT attack. Probing is an attack where the attacker attempts to gather information about the network to exploit its protection by sending an ipsweep-ping to several hosts to discover the IP address of the target and scan for ports to discover the services of the host. Reconnaissance attacks are the fifth most frequent IoT attack. In reconnaissance attacks the attacker collects

information for the system in order to observe it. Table 9 below summarizes the IoT attacks according to the layers.

Table 9. IoT attacks according to the layers.

IoT Attack	Layer
DoS	*Network layer* and application layer.
U2R	*Application layer.*
R2L	*Network layer* and perception layer.
Probing	*Network layer.*
Reconnaissance	*Network layer.*
wormhole	*Processing layer.*
DDoS	*Network layer* and application layer.
backdoor	*Application layer.*
analysis	*Application layer.*
generic	Application Layer.
fuzzers	*Network layer* and perception layer.
shellcode	Processing layer.
sinkhole	*Network layer.*
blackhole	Perception layer.
hello flooding	*Network layer.*
SQL injection	Processing layer.
ARP cache poisoning	*Network layer.*
Malformed packets	Application layer.
Exploits	*Network layer.*
Scanning	*Network layer.*

Other IoT attacks that have rarely been considered in primary studies are wormhole (4.8%), distributed denial-of-service (DDoS) (4.8%), backdoor (4.1%), analysis (3.4%), generic (2.7%), fuzzers (2.7%), shellcode (2.7%), sinkhole (2%), blackhole (2%), hello flooding (1.4%), SQL injection (1.4%), ARP cache poisoning (1.4%), malformed packets (1.4%), exploits (1.4%), and scanning (1.4%) attacks.

3.4. Independent Variables

The independent variables used in machine learning models, also called predictors or features, play important roles in enabling good performance in the detection of IoT attacks. Some primary studies use techniques to decrease the dimensions of the dataset from a massive number of features to a small number. Shengchu et al. [43] and Hamed et al. [46] used principal component analysis (PCA) to decrease the dimensions of a dataset from a large number of features to a small number. Monika et al. [55] used NSGA-ii-aJG to decrease the dimensions of a dataset from a large number of features to a small number. Shubhra et al. [60] used an information gain-based intrusion detection system (IGIDS) to select the most relevant features from the original IDS datasets. The independent variables used in the IoT attack detection model depend on the type of IoT attack detected and the datasets used, such as public datasets or IoT device testbed datasets. Table 10 below summarizes the primary studies with information on public datasets used, IoT attack type, and feature used in proposed model for IoT attack detection. Table 11 below summarizes the primary studies used IoT device testbed datasets with information on feature used in proposed model for IoT attack detection.

Table 10. Features considered in primary studies (public datasets).

Datasets	IoT Attack Type	Features
NSL-KDD	DoS	back, teardrop, Neptune, land, smurf, and pod.
	U2R	buffer overflow, perl, rootkit, and load module.
	Probing	Satan, ipsweep, portsweep, and nmap.
	R2L	multihop, warezmaster, FTP write, guess password, phf, spy, imap, and warezclient.
UNSW-NB15	Fuzzers, analysis, reconnaissance, backdoors, generic, DoS, exploits, worms, and shellcode	destination, service, source mean, source byte, source to destination time, mean size, source inter-packet arrival time, data transferred, protocol, number of connections, and number of flows.
KDDCUPP99	Probing	nmap, satan, ipsweep, saint, portsweep, and mscan.
	U2R	perl, sqlattack, Httptunnel, buffer_overflow, ps, rootkit, xterm, and loadmodule.
	R2L	Xlock, xsnoop, phf, snmpguess, warezclient, named, warezmaster, tp_write, spy, guess_passwd, imap, snmpgetattack, worm, and multihop.
	DoS	Neptune, back, teardrop, mailbomb, land, processtable, apache2, udpstorm, smurf, and pod.
CICIDS2017	DDoS	Source, time stamps, and destination IP addresses.
RPL-NIDDS17	Sinkhole, Sybil, Clone ID, Blackhole, Hello Flooding, Selective Forwarding, and Local Repair attacks	Destination IP address, protocols used, time of the attack, and size of packets transmitted.
BoT-IoT	probing, DOS, and DDOS	frame-related fields, ARP-related fields, IP-related fields, TCP-related fields, and UDP-related fields.

Table 11. Features considered in primary studies (IoT device testbed datasets).

S.No	Features
S1	destination IP address, protocols used, time of the attack, and size of packets transmitted.
S3	Frame information and packet type.
S6	Safe distance between any two neighboring routers.
S7	Flags, Ip_len, TCP4_flood, UDP_Flood, TCP6_Flood, UDP6_Low, and IP6_plen.
S8	Packet receiving rate and consumed energy.
S10	mean flow (mean), destination, source bytes, source packets, source port, and total load.
S17	two classes: connection features (e.g., duration of connection, packets per second, average size of data message, and data rate) and traffic features (e.g., active connections on a specific port, active connections on all hosts, rate of active connections on a specific host, rate of active connections for a service, and active connections on a specific host port).
S18	Data packets sent, packets forwarded, packets dropped, announcements received, and data packets received.
S20	bandwidth consumption, source of requests, number of failed authentication attempts, number of sent requests, and device usage at different periods.
S21	the number of bytes in acknowledgment response packets, the number of bytes in command packets, and inter-packet time interval.
S22	level, time, source IP, and packet type.
S24	source bytes, average packet size of traffic, and destination.
S26	request identifier, destination, and response status code.
S30	sequence number, destination port, and window size.
S31	window size, sequence number, and destination port.
S32	duration of connection, rate transmission, and destination.

Table 11. *Cont.*

S.No	Features
S33	transmission rate, reception rate, source IP, and destination.
S38	packets forwarded, packets dropped, data packets sent, and announcements received.
S41	destination IP address of the packet, sequence number for the packet, time, source IP address of the packet, protocol, length of the packet, and info.
S47	Duration, total forward packet, total backward packet, total length backward packet, total length forward packet, and idle minimum time.

3.5. Evaluation Metrics

Detect attacks should be evaluated in real time to assess their effectiveness and efficiency. The primary studies we reviewed used various strategies to evaluate the efficiency of their proposed approach. Figure 5 shows the percentages of specific evaluation metrics used in the primary studies. Numerical measures and graphical measures are two types of measurement. Numerical measures consist of precision, accuracy, F-measure, and others, whereas graphical measures consist of ROC curves, etc.

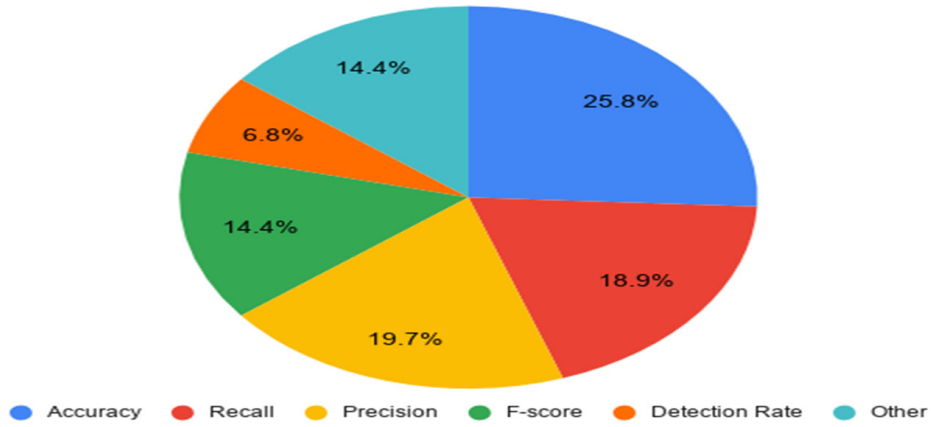

Figure 5. Evaluation metrics used in primary studies.

Accuracy was frequently used in the primary studies. Accuracy can be described as the number of IoT attacks that are correctly detected divided by the number of IoT attacks. The second most commonly used performance measure for the identification of IoT attacks is recall. This measurement relates to the quantity of IoT attack classes correctly predicted among the actual IoT attack classes. Precision is the third most commonly used evaluation metric, and it measures the correctness of the model. F-measure is the fourth most commonly used evaluation metric, and it shows the trade-off between the performances of a classifier. The detection rate is the fifth most commonly used evaluation metric, and it indicates the efficiency of a classifier with respect to its ability to detect malicious behaviors.

3.6. DevSecOps

DevOps is the process of continuously improving software products through rapid release cycles, global automation of integration and delivery pipelines, and close collaboration between teams [81]. Securing DevOps helps organizations operate securely and protect the data their customers entrust them with. DevSecOps represents a cultural solution for improving and accelerating business value delivery by effectively coordinating development, security, and operations [82]. If cyber security is achieved after completion of development, systems shall be developed with vulnerabilities that are impossible to solve. Security teams must exchange expertise and supply resources for operation and

development teams that fit systems and applications [83]. If the detection models applied DevSecOps pipelines in development processes for IoT devices, they were more secure.

Few studies dynamically generated and configured IoT system infrastructure management using DevSecOps. Athanasios et al. [84] proposed a system for automatic lifecycle management of IoT applications that require cellular network connectivity. This system uses DevOps pipeline by automating the deployment of IoT application based on the information retrieved from the monitoring infrastructure (CPU, memory status, and network). Jessica et al. [33] addressed the formalization of feedback processes from operations to IoT system development. Security teams use the continuous and fast process from Ops to Dev to instantiate IoT's self-service cyber security management systems to enforce security in a DevOps environment. Ramón et al. [85] proposed an architectural model of a distributed IoT system and continuous delivery (CD) of customized Software as a Service (SaaS) updates at the IoT Edge. The proposed model automated building, deployment, and testing of software updates for edge devices. Miguel et al. [86] addressed the formalization of continuous and fast feedback to detect problems in an IoT system in order to fix them.

4. Discussion

In this study we reviewed 49 journal papers on IoT attack detection that were published from 2016 to 2020. We have provided a summary of IoT attack detection models and identified the scope of the development models. We collected all available papers in various digital libraries.

There are different features for each IoT layer, so there are multiple threats for each layer. Most IoT attacks occur at the network layer according to the literature. IoT attacks can be detected in any layer of IoT architecture. The binary class classification is commonly used in IoT attack detection models. Inputs are labeled in binary class classification as an attack or as benign. Some studies use multiclass classifications not only to recognize attacks, but to also identify their type.

Following the research questions defined above in Section 2.1, the first question is related to the type of datasets that researchers often use to construct a detection model in primary studies. Most primary studies used IoT device testbed datasets, and others used public datasets. The NSL-KDD, UNSW-NB15, and KDDCUP99 repositories were found to be the most frequently used datasets among researchers. However, public datasets have some quality issues, which can lead to poor detection results. However, studies that use data from real IoT device traffic enhance the effectiveness of ML techniques.

The second question is related to machine learning techniques that are often used for building detection models, and the NN has been widely used in IoT attack detection models. However, with standard CPUs, NNs are computationally more time-consuming and costly. EL techniques have rarely been used in IoT attack detection models. SVMs are not recommended for large datasets, as the training takes a long time. Some researchers have proposed hybrid frameworks [29,54,56,64,68]. Some studies have proposed distributed attack detection [46,49,50], which has achieved better attack detection than centralized algorithms.

The third question is related to IoT attacks detected in the proposed model, where DoS is the most commonly detected type of attack in the primary studies. DoS is popular because it aims to misuse the available resources in a communication network and stop services used by users. Therefore, researchers need to purpose this model for IoT attack detection.

The fourth question is related to the independent variables used in primary studies, which depend on the type of IoT attack detected and the datasets used, such as public datasets or IoT device testbed datasets. NSL-KDD datasets have 41 features, such as service, duration, flag, destination bytes, protocol, source bytes, etc. UNSW-NB15 datasets have nine attacks and 49 features, such as destination, service, source mean, source byte, etc. KDDCUP99 have three attacks and 41 features, such as nmap, satan, ipsweep, saint, portsweep, mscan, etc.

The fifth question is related to evaluation metrics, of which accuracy is the most commonly used. Accuracy is popular because it is used to measure the ratio of correct predictions over the total number of instances evaluated.

The last question is related to identifying whether existing models or systems are monitoring an infrastructure that is created and configured automatically for IoT systems using DevSecOps pipelines. Few studies have analyzed device log traces from IoT devices to identify IoT attacks and monitor infrastructure using DevSecOps pipelines.

In this study, we also raised several challenges when it comes to IoT attack detection and included an overview of the work that can be performed to overcome these challenges. The first challenge that researchers have discovered is using public datasets such as NSL-KDD, UNSW-NB15, and KDDCUP99 in IoT attack detection models. Public datasets have some quality issues, which can lead to poor detection in IoT attack models. We recommend applying some data preprocessing and data cleaning techniques to improve the quality of public datasets.

Another challenge relates to building IoT attack models. More research on the detection of IoT attacks should be performed using ML techniques to achieve generalizable results since there are very few studies comparing various ML algorithms. Researchers can apply other approaches such as ensemble learning (EL) algorithms and other classifiers to detect the IoT attacks. A few studies have used hybrid frameworks, which achieved good performance and high detection rates compared to the use of individual machine learning algorithms. Thus, we recommend the continued use of hybrid frameworks for the improved detection of IoT attacks. Moreover, distributed attack detection algorithms have achieved better attack detections than centralized algorithms, so we recommend using distributed attack models.

Another challenge is that there was only one study that dynamically generated and configured IoT system infrastructure management using DevSecOps. Jessica Diaz [33] addresses the formalization of feedback processes from operations to IoT system development. This infrastructure was implemented following good DevOps practices. It was automated by configuration files and scripts (monitoring as code), and its deployment was simplified by virtualization and containerization technologies and versioned (GitHub). We recommend advanced monitoring infrastructure configurations using methods based on software pipelines and the use of machine learning techniques for advanced supervision and monitoring.

Study Limitations

Our review has many limitations. First, the search keywords selected and time of publication (last 5 years) limit this study. Second, it utilized few electronic sources. Moreover, this study discussed only English papers and we cannot guarantee to have used all the good studies for our review. Third, the data are provided by private security companies, such as McAffe and Symantec. It is common for these companies to not publish scientific papers.

5. Conclusions and Future Work

In this study, we reviewed the performance of IoT attack detection models that use machine learning techniques to analyze and evaluate attacks. We identified 49 primary studies between 2016 and 2020 after a comprehensive investigation following an organized process. We summarized these primary studies based on the datasets, ML techniques, types of IoT attack, independent variables, evaluation metrics, and monitoring infrastructure via DevSecOps pipelines. We summarize the main findings as follows:

- Most primary studies used IoT device testbed datasets, and others used public datasets. NSL-KDD, UNSW-NB15, and KDDCUP99 repositories were found to be the most frequently used datasets among researchers.
- BN, DT, NN, clustering, SVM, FS, and EL were the ML techniques used in primary studies, and NNs were the most widely used technique for IoT attack detection.

- DOS, U2R, and R2L attacks were most widely considered in the primary studies based on the results we obtained.
- Accuracy, recall, and precision were the most widely used evaluation metrics in the primary studies.
- Few studies analyzed device log traces from IoT devices to identify IoT attacks and monitor infrastructure using DevSecOps pipelines.

For future studies on IoT attack detection using machine learning techniques, the following are recommended:

- More data preprocessing and data cleaning techniques should be applied to improve the quality of public datasets.
- Using data from real IoT device traffic will enhance the effectiveness of ML techniques.
- The performance of IoT attack detection models should continue to be enhanced through integration with other algorithms.
- Infrastructure configuration should continue to be monitored using methods based on software pipelines.
- Machine learning techniques should be used for advanced supervision and monitoring.

Author Contributions: Conceptualization, A.B. and H.F.; methodology, A.S.; software, A.A.; validation, A.B., H.F. and L.E.; formal analysis, H.F.; investigation, A.B.; resources, H.F.; data curation, A.S.; writing—original draft preparation, A.S.; writing—review and editing, A.A.; visualization, A.S.; supervision, L.E.; project administration, H.F.; funding acquisition, A.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Not applicable.

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

References

1. Mehta, R.; Sahni, J.; Khanna, K. Internet of things: Vision, applications and challenges. *Procedia Comput. Sci.* **2018**, *132*, 1263–1269. [CrossRef]
2. ISO/IEC International Standard 20924. Information Technology-Internet of Things-Definition and Vocabulary. 2018. Available online: https://www.iso.org/standard/69470.html. (accessed on 30 March 2021).
3. Yang, Z.; Liang, B.; Ji, W. An Intelligent end-edge-cloud architecture for visual iot assisted healthcare systems. *IEEE Internet Things J.* **2021**, *8*. [CrossRef]
4. Jiang, X.; Zhang, H.; Yi, E.A.B.; Raghunathan, N.; Mousoulis, C.; Chaterji, S.; Dimitrios, P.; Shakouri, A.; Bagchi, S. Hybrid low-power wide-area mesh network for iot applications. *IEEE IoT J.* **2020**, *8*, 901–915. [CrossRef]
5. Jadhav, A.R.; MPR, S.K.; Pachamuthu, R. Development of a novel iot-enabled power-monitoring architecture with real-time data visualization for use in domestic and industrial scenarios. *IEEE Trans. Instrum. Measure.* **2020**, *70*, 1–14. [CrossRef]
6. Statista Report. Available online: https://www.statista.com/statistics/471264/iotnumber-of-connected-devices-worldwide (accessed on 2 January 2021).
7. Alam, T. A reliable communication framework and its use in internet of things (IoT). *Int. J. Sci. Res. Comp. Sci. Eng. Inf. Technol.* **2018**, *3*. [CrossRef]
8. Cyberattack Knocks Out Access to Websites. Available online: https://www.wsj.com/articles/denial-of-service-web-attack-affects-amazon-twitter-others-1477056080 (accessed on 2 January 2021).
9. Sajjad, S.M.; Yousaf, M.; Afzal, H.; Muftid, M.R. eMUD: Enhanced manufacturer usage description for IoT botnets prevention on home WiFi routers. *IEEE Access* **2020**, *8*, 164200–164213. [CrossRef]
10. Chen, S.; Xu, H.; Liu, D.; Hu, B.; Wang, H. A vision of IoT: Applications, challenges, and opportunities with china perspective. *IEEE IoT J.* **2014**, *1*, 349–359.
11. Doshi, K.; Yilmaz, Y.; Uludag, S. Timely detection and mitigation of stealthy DDoS attacks via IoT networks. *IEEE Trans. Dependable Secur. Comput.* **2021**, *1*. [CrossRef]
12. McAfee Labs COVID-19 Threats Report. Available online: https://www.mcafee.com/enterprise/en-us/assets/reports/rp-quarterlythreats-july-2020.pdf (accessed on 28 March 2021).

13. Singh, S.; Hosen, A.S.; Yoon, B. Blockchain Security Attacks, Challenges, and Solutions for the Future Distributed IoT Network. *IEEE Access* **2021**, *9*, 13938–13959. [CrossRef]
14. Chowdhury, M.R.; Tripathi, S.; De, S. Adaptive multivariate data compression in smart metering Internet of Things. *IEEE Trans. Ind. Inform.* **2020**, *17*, 1287–1297. [CrossRef]
15. Makkar, A.; Garg, S.; Kumar, N.; Hossain, M.S.; Ghoneim, A.; Alrashoud, M. An efficient spam detection technique for IoT devices using machine learning. *IEEE Trans. Ind. Inform.* **2020**, *17*, 903–912. [CrossRef]
16. Darko, A.; Vrček, N. Machine learning for the Internet of Things security: A systematic review. In Proceedings of the International Conference on Software Technologies (ICSOFT), Porto, Portugal, 26–28 July 2018; pp. 563–570. [CrossRef]
17. Kavianpour, S.; Shanmugam, B.; Azam, S.; Zamani, M.; Samy, G.N.; De Boer, F. A systematic literature review of authentication in Internet of Things for heterogeneous devices. *J. Comput. Netw. Commun.* **2019**, *2019*, 1–14. [CrossRef]
18. 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 IoT J.* **2019**, *6*, 8182–8201. [CrossRef]
19. Aly, M.; Khomh, F.; Haoues, M.; Quintero, A.; Yacout, S. Enforcing security in Internet of Things frameworks: A systematic literature review. *IoT* **2019**, *6*, 100050. [CrossRef]
20. Ihsan, A.; Abdelmuttlib, I.A.A.; Almogren, A.; Raza, M.A.; Shah, S.A.; Khan, A.; Gani, A. Systematic literature review on IoT-based botnet attack. *IEEE Access* **2021**, *8*, 212220–212232.
21. Hinderks, A.; José, F.; Mayo, D.; Thomaschewski, J.; Escalona, M.J. An SLR-tool: Search process in practice: A tool to conduct and manage systematic literature review (SLR). In Proceedings of the 2020 IEEE/ACM 42nd International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), Seoul, Korea, 5–11 October 2020; pp. 81–84.
22. Kitchenham, B.; Charters, S. *Guidelines for Performing Systematic Literature Reviews in Software Engineering*; School of Computer Science and Mathematics, Keele University: Staffordshire, UK, 2007.
23. Anthi, E.; Williams, L.; Słowińska, M.; Theodorakopoulos, G.; Burnap, P. Pulse: An Adaptive Intrusion Detection for the Internet of Things, Living in the Internet of Things: Cybersecurity of the IoT. In Proceedings of the Living in the Internet of Things: Cybersecurity of the IoT—2018, London, UK, 28–29 March 2018; pp. 1–4.
24. Anthi, E.; Williams, L.; Słowińska, M.; Theodorakopoulos, G.; Burnap, P. A supervised intrusion detection system for smart home IoT devices. *IEEE IoT J.* **2019**, *6*, 9042–9053. [CrossRef]
25. Prachi, S. *ML*-IDS: A Machine Learning Approach to Detect Wormhole Attacks in Internet of Things. In Proceedings of the Intelligent Systems Conference (IntelliSys), London, UK, 7–8 September 2017; pp. 234–240.
26. Bakhtiar, F.A.; Pramukantoro, E.S.; Nihri, H. A lightweight ids based on J48 algorithm for detecting dos attacks on IoT Middleware. In Proceedings of the IEEE 1st Global Conference on Life Sciences and Technologies (LifeTech), Osaka, Japan, 12–14 March 2019; pp. 41–42.
27. Yahyaoui, A.; Abdellatif, T.; Attia, R. Hierarchical anomaly-based intrusion detection and localization in IoT. In Proceedings of the 15th International Wireless Communications & Mobile Computing Conference (IWCMC), Tangier, Morocco, 24–28 June 2019; pp. 108–113.
28. Zolanvari, M.; Teixeira, M.A.; Gupta, L.; Khan, K.M.; Jain, R. Machine learning-based network vulnerability analysis of industrial Internet of Things. *IEEE Internet Things J.* **2019**, *6*, 6822–6834. [CrossRef]
29. Bhatt, P.; Morais, A. HADS: Hybrid anomaly detection system for IoT environments. In Proceedings of the International Conference on Internet of Things, Embedded Systems and Communications (IINTEC), Hamammet, Tunisia, 20–21 December 2018; pp. 191–196.
30. Ioannou, C.; Vassiliou, V. Classifying security attacks in IoT networks using supervised learning. In Proceedings of the 2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), Santorini, Greece, 29–31 May 2019; pp. 652–658. [CrossRef]
31. Bhunia, S.S.; Gurusamy, M. Dynamic attack detection and mitigation in IoT using SDN. In Proceedings of the 27th International Telecommunication Networks and Applications Conference (ITNAC), Melbourne, VIC, Australia, 22–24 November 2017; pp. 1–6.
32. Nobakht, M.; Sivaraman, V.; Boreli, R. A host-based intrusion detection and mitigation framework for smart home IoT using OpenFlow. In Proceedings of the 11th International Conference on Availability, Reliability and Security (ARES), Salzburg, Austria, 31 August–2 September 2016; pp. 147–156.
33. Díaz, J.; Pérez, J.E.; Lopez-Peña, M.A.; Mena, G.A.; Yagüe, A. Self-service cybersecurity monitoring as enabler for devsecops. *IEEE Access* **2019**, *7*, 100283–100295. [CrossRef]
34. Bhatia, R.; Benno, S.; Esteban, J.; Lakshman, T.V.; Grogan, J. Unsupervised machine learning for net-work-centric anomaly detection in IoT. In Proceedings of the 3rd ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks, New York, NY, USA, 1–5 December 2019.
35. Chaabouni, N.; Mosbah, M.; Zemmari, A.; Sauvignac, C. *An Intrusion Detection System for the OneM2M Service Layer Based on Edge Machine Learning*; Springer: Cham, Switzerland, 2019.
36. Hashimoto, N.; Ozawa, S.; Ban, T.; Nakazato, J.; Shimamura, J. A darknet traffic analysis for IoT malwares using association rule learning. *Procedia Comput. Sci.* **2018**, *144*, 118–123. [CrossRef]
37. Ozawa, S.; Ban, T.; Hashimoto, N.; Nakazato, J.; Shimamura, J. A study of IoT malware activities using association rule learning for darknet sensor data. *Int. J. Inform. Secur.* **2020**, *19*, 83–92. [CrossRef]

38. Maleh, Y. Machine learning techniques for IoT intrusions detection in aerospace cyber-physical systems. In *Machine Learning and Data Mining in Aerospace Technology*; Springer: Cham, Switzerland, 2019.
39. Thamilarasu, G.; Chawla, S. Towards deep-learning-driven intrusion detection for the Internet of Things. *Sensors* **2019**, *19*, 1977. [CrossRef]
40. Ioannou, C.; Vassiliou, V. Experimentation with local intrusion detection in IoT networks using supervised learning. In Proceedings of the 16th International Conference on Distributed Computing in Sensor Systems (DCOSS), Marina del Rey, CA, USA, 25–27 May 2020; pp. 423–428.
41. Liu, Z.; Thapa, N.; Shaver, A.; Kaushik, R.; Xiaohong, Y.; Khorsandroo, S. Anomaly detection on loT network intrusion using machine learning. In Proceedings of the International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD), Durban, South Africa, 6–7 August 2020; pp. 1–5.
42. Pecori, R.; Tayebi, A.; Vannucci, A.; Veltri, L. IoT Attack detection with deep learning analysis. In Proceedings of the International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–8.
43. Liang, C.; Shanmugam, B.; Azam, S.; Jonkman, M.; De Boer, F.; Narayansamy, G. Intrusion detection system for Internet of Things based on a machine learning approach. In Proceedings of the International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), Vellore, India, 30–31 March 2019.
44. Illy, P.; Kaddoum, G.; Moreira, C.M.; Kaur, K.; Garg, S. Securing fog-to-things environment using intrusion detection system based on ensemble learning. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Marrakesh, Morocco, 1 April 2019; pp. 1–7.
45. Pajouh, H.H.; Javidan, R.; Khayami, R.; Dehghantanha, A.; Choo, K.K.R. A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks. *IEEE Trans. Emerg. Top. Comp.* **2016**, *7*, 314–323. [CrossRef]
46. Abeshu, A.; Chilamkurti, N. Deep learning: The frontier for distributed attack detection in fog-to-things computing. *IEEE Comm. Mag.* **2018**, *56*, 169–175. [CrossRef]
47. Shalaginov, A.; Semeniuta, O.; Alazab, M. MEML: Resource-Aware MQTT-Based Machine Learning for Network Atack Detection on IoT Edge Devices. In Proceedings of the 12th IEEE/ACM International Conference on Utility and Cloud Computing Companion (UCC'19 Companion), Auckland, New Zealand, 1–5 December 2019.
48. Rezvy, S.; Petridis, M.; Lasebae, A.; Zebin, T. *Intrusion Detection and Classification with Au-toencoded Deep Neural Network*; Springer: Cham, Switzerland, 2019.
49. Rathore, S.; Park, J.Y. Semi-supervised learning based distributed attack detection framework. *Appl. Soft Comp.* **2018**, *72*, 79–89. [CrossRef]
50. Diro, A.A.; Chilamkurti, N. Distributed attack detection scheme using deep learning approach for Internet of Things. *Future Gener. Comp. Syst.* **2018**, *82*, 761–768. [CrossRef]
51. Fenanir, S.; Semchedine, F.; Baadache, A. A machine learning-based lightweight intrusion detection system for the Internet of Things. *Rev. d'Intelligence Artif.* **2019**, *33*, 203–211. [CrossRef]
52. Verma, A.; Ranga, V. Machine learning based intrusion detection systems for IoT applications. *Wireless Pers Commun.* **2019**, *111*, 2287–2310. [CrossRef]
53. Zhang, J.; Gong, L.R.; Yu, K.; Qi, X.; Wen, Z.; Hua, Q. 3D reconstruction for super-resolution CT images in the Internet of health things using deep learning. *IEEE Access* **2020**, *8*, 121513–121525. [CrossRef]
54. Taghavinejad, S.M.; Taghavinejad, M.; Shahmiri, L.; Zavvar, M.; Zavvar, M.H. Intrusion detection in IoT-based smart grid using hybrid decision tree. In Proceedings of the 6th International Conference on Web Research (ICWR), Tehran, Iran, 22–23 April 2020; pp. 152–156.
55. Al-Emadi, S.; Al-Mohannadi, A.; Al-Senaid, F. Using deep learning techniques for network intrusion detection. In Proceedings of the IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), Doha, Qatar, 2–5 February2020; pp. 171–176.
56. de Souzaa, C.A.; Becker Westphalla, C.; Bobsin Machadob, R.; Mangueira Sobrala, B.; dos Santos Vieirab, G. Hybrid approach to intrusion detection in fog-based IoT environments. *Comp. Netw.* **2020**, *180*, 107417. [CrossRef]
57. Rani, D.; Kaushal, N.C. Supervised machine learning based network intrusion detection system for Internet of Things. In Proceedings of the 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 1–3 July 2020; pp. 1–7.
58. Zhao, S.; Li, W.; Zia, T.; Zomaya, A.Y. A dimension reduction model and classifier for anomaly-based intrusion detection in Internet of Things. In Proceedings of the 2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech), Orlando, FL, USA, 6–10 November 2017; pp. 836–843.
59. Indre, I.; Lemnaru, C. Detection and prevention system against cyber attacks and botnet malware for information systems and internet of things. In Proceedings of the 12th International Conference on Intelligent Computer Communication and Processing (ICCP), Cluj-Napoca, Romania, 8–10 September 2016; pp. 175–182.
60. Dwivedi, S.; Vardhan, M.; Tripathi, S. Distributed denial-of-service prediction on IoT framework by learning techniques. *Open Comput. Sci.* **2020**, *10*, 220–230. [CrossRef]

61. Bipraneel, R.; Cheung, H. A deep learning approach for intrusion detectionin internet of things using BI-directional long short-term memory recurrent neural network. In Proceedings of the 28th International Telecommunication Networks and Applications Conference (ITNAC), Sydney, NSW, Australia, 21–23 November 2018.
62. Hanif, S.; Ilyas, T.; Zeeshan, M. Intrusion detection in IoT using artificial neural networks on UNSW-15 Dataset. In Proceedings of the 2019 IEEE 16th International Conference on Smart Cities: Improving Quality of Life Using ICT & IoT and AI (HONET-ICT), Charlotte, NC, USA, 6–9 October 2019; pp. 152–156.
63. Latif, S.; Idrees, Z.; Zou, Z.; Ahmad, J. DRaNN: A deep random neural network model for intrusion detection in industrial IoT. In Proceedings of the International Conference on UK-China Emerging Technologies (UCET), Glasgow, UK, 20–21 August 2020; pp. 1–4.
64. Chkirbene, Z.; Eltanbouly, S.; Bashendy, M.; Alnaimi, N.; Erbad, A. Hybrid machine learning for network anomaly intrusion detection. In Proceedings of the IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), Doha, Qatar, 2–5 February 2020; pp. 163–170.
65. Verma, A.; Ranga, V. ELNIDS: Ensemble Learning based Network Intrusion Detection System for RPL based Internet of Things. In Proceedings of the 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), Ghaziabad, India, 18–19 April 2019; pp. 1–6.
66. Shakhov, V.; Ullah Jan, S.; Ahmed, S. On Lightweight method for intrusions detection in the Internet of Things. In Proceedings of the International Black Sea Conference on Communications and Networking (BlackSeaCom), Sochi, Russia, 3–6 June 2019; pp. 1–5.
67. Ge, M.; Fu, X.; Syed, N.; Baig, Z.; Teo, G.; Robles-Kelly, A. Deep learning-based intrusion detection for IoT networks. In Proceedings of the 24th Pacific Rim International Symposium on Dependable Computing (PRDC), Kyoto, Japan, 1–3 December 2019; pp. 256–266.
68. Hosseini, M.; Borojeni, H.R.S. A hybrid approach for anomaly detection in the Internet of Things. In Proceedings of the Proceedings of the International Conference on Smart Cities and Internet of Things-SCIOT'18, Mashhad, Iran, 1–3 September 2018.
69. Latif, S.; Zou, Z.; Idrees, Z.; Ahmad, J. A novel attack detection scheme for the industrial internet of things using a lightweight random neural network. *IEEE Access* **2020**, *8*, 89337–89350. [CrossRef]
70. Roopak, M.; Tian, G.Y.; Chambers, J. Intrusion detection system against ddos attacks in loT networks. In Proceedings of the 10th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 6–8 January 2020; pp. 0562–0567.
71. Alzahrani, H.; Abulkhair, M.; Alkayal, E. A multi-class neural network model for rapid detection of IoT botnet attacks. *Int. J. Adv. Comp. Sci. Appl.* **2020**. [CrossRef]
72. Alsaedi, A.; Moustafa, N.; Tari, Z.; Mahmood, A.; Anwar, A. TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems. *IEEE Access* **2020**, *8*, 165130–165150. [CrossRef]
73. Yahyaoui, A.; Abdellatif, T.; Yangui, S.; Attia, R. "READ-IoT: Reliable anomalies and events detection framework for the Internet of Things. *IEEE Access* **2021**, *9*, 24168–24186. [CrossRef]
74. Wang, Z.; Zeng, Y.; Liu, Y.; Li, D. Deep belief network integrating improved kernel-based extreme learning machine for network intrusion detection. *IEEE Access* **2021**, *9*, 16062–16091. [CrossRef]
75. Maseer, Z.K.; Yusof, R.; Bahaman, N.; Mostafa, S.A.; Foozy, C.F.M. Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset. *IEEE Access* **2021**, *9*, 22351–22370. [CrossRef]
76. Pu, G.; Wang, L.; Shen, J.; Dong, F. A hybrid unsupervised clustering-based anomaly detection method. *Tsinghua Sci. Technol.* **2021**, *26*, 146–153. [CrossRef]
77. Liu, Y.; Wang, J.; Li, J.; Song, H.; Yang, T.; Niu, S.; Ming, Z. Zero-bias deep learning for accurate identification of Internet-of-Things (IoT) devices. *IEEE Internet Things J.* **2021**, *8*, 2627–2634. [CrossRef]
78. Nair, R.; Sharma, P.; Kumar Singh, D. *Security Attacks in Internet of Things*; Wiley Online Library Publishing: Hoboken, NJ, USA, 2020; Chapter 14; pp. 237–261. [CrossRef]
79. Sharma, P.; Kherajani, M.; Jain, D.; Patel, D. A Study of Routing Protocols, Security Issues and Attacks in Network Layer of Internet of Things Framework. In Proceedings of the 2nd International Conference on Data, Engineering and Applications (IDEA), Bhopal, India, 28–29 February 2020; pp. 1–6.
80. Raghuprasad, A.; Padmanabhan, S.; Babu, M.A.; Binu, P.K. Security analysis and prevention of attacks on IoT devices. In Proceedings of the International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 28–30 July 2020; pp. 0876–0880.
81. Rafi, S.; Yu, W.; Akbar, M.A.; AlSanad, A.; Gumaei, A. Prioritization based taxonomy of DevOps security challenges using PROMETHEE. *IEEE Access* **2020**, *8*, 105426–105446. [CrossRef]
82. Mohan, V.; Othmane, L.B. SecDevOps: Is it a marketing buzzword?-Mapping research on security in DevOps. In Proceedings of the 11th International Conference on Availability, Reliability and Security (ARES), Salzburg, Austria, 31 August–2 September 2016; pp. 542–547.
83. Carter, K. Francois Raynaud on DevSecOps. *IEEE Softw.* **2017**, *34*, 93–96. [CrossRef]

84. Karapantelakis, A.; Liang, H.; Wang, K.; Vandikas, K.; Inam, R.; Fersman, E.; Mulas-Viela, I.; Seyvet, N.; Giannokostas, V. DevOps for IoT Applications using cellular networks and Cloud. In Proceedings of the IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud), Vienna, Austria, 22–24 August 2016; pp. 340–347.
85. Lopez-Viana, R.; Diaz, J.; Diaz, V.H.; Martinez, J.-F. Continuous delivery of customized SaaS edge applications in highly distributed IoT systems. *IEEE Internet Things J.* **2020**, *7*, 10189–10199. [CrossRef]
86. López-Peña, M.A.; Díaz, J.; Pérez, J.E.; Humanes, H. DevOps for IoT systems: Fast & continuous monitoring feedback of system availability. *IEEE Internet Things J.* **2020**, *7*, 10695–10707. [CrossRef]