electronics

MDPI

Article

A Novel Ensemble of Hybrid Intrusion Detection System for Detecting Internet of Things Attacks

Ansam Khraisat *, Iqbal Gondal, Peter Vamplew, Joarder Kamruzzaman and Ammar Alazab

Internet Commerce Security Laboratory, Federation University Australia, Mount Helen 3350, Australia; iqbal.gondal@federation.edu.au (I.G.); p.vamplew@federation.edu.au (P.V.); joarder.kamruzzaman@federation.edu.au (J.K.); aalazab@mit.edu.au (A.A.)

* Correspondence: a.khraisat@federation.edu.au

Received: 5 September 2019; Accepted: 18 October 2019; Published: 23 October 2019

Abstract: The Internet of Things (IoT) has been rapidly evolving towards making a greater impact on everyday life to large industrial systems. Unfortunately, this has attracted the attention of cybercriminals who made IoT a target of malicious activities, opening the door to a possible attack to the end nodes. Due to the large number and diverse types of IoT devices, it is a challenging task to protect the IoT infrastructure using a traditional intrusion detection system. To protect IoT devices, a novel ensemble Hybrid Intrusion Detection System (HIDS) is proposed by combining a C5 classifier and One Class Support Vector Machine classifier. HIDS combines the advantages of Signature Intrusion Detection System (SIDS) and Anomaly-based Intrusion Detection System (AIDS). The aim of this framework is to detect both the well-known intrusions and zero-day attacks with high detection accuracy and low false-alarm rates. The proposed HIDS is evaluated using the Bot-IoT dataset, which includes legitimate IoT network traffic and several types of attacks. Experiments show that the proposed hybrid IDS provide higher detection rate and lower false positive rate compared to the SIDS and AIDS techniques.

Keywords: IoT; network; security; anomaly detection; zero-day malware; intrusion; intrusion detection system

1. Introduction

Internet of Things (IoT) is an interconnected system of devices that facilitate seamless information exchange between physical devices. These devices could be medical and healthcare devices, driverless vehicles, industrial robots, smart TVs, wearables and smart city infrastructures, and they can be remotely monitored and regulated [1,2]. IoT devices are expected to become more prevalent than mobile devices and will have access to the most sensitive information such as personal information [3]. This will result in increasing attack surface area and probabilities of attacks will increase. For instance, 'Mirai' is a botnet that mounted a Distributed Denial of Service (DDoS) attack that left much of the network unapproachable [4].

Due to the significance of IoT devices in our daily lives, it is crucial to develop IoT intelligent IDS capable of detecting both pre-known and zero-day attacks. As IoT devices are part of infrastructure, it makes them a target of cyber-attacks. Symantec reported a 600% increase in attacks against the IoT platforms in 2018 [5], which means that attackers are aiming to exploit the connected nature of these devices.

Intrusion Detection System (IDS) technology has originally been developed for traditional networks, and therefore, the current techniques IDSs for IoT are insufficient to detect different types of attacks for the following reasons [6]. First, the current IDS protect against known security threats, which means they are easily defeated by the new kinds of intrusions by attackers, as they can evade the traditional IDS [7]. For instance, the increased volume of DDoS attacks uses techniques that spoof

source IP addresses to hide attacks, so it becomes undetectable by the traditional IDS. Second, IoT specific features present a challenge for creating IDS. IoT devices are huge in number and need to host IDS agents; furthermore, low storage and computational capacity of IoT devices impose constraints on how IDS systems can be implemented. Third, another important issue is the characteristic associated with the IoT network design. In the traditional networks, the computer system is completely connected to specific computer nodes that are responsible for sending packets to the endpoints. In contrast, the IoT ecosystem communicates with numerous sensors and actuators to accomplish several monitoring and control tasks. IoT devices have significantly more varieties and type of networks than traditional networks. Therefore, applying traditional IDS detection system to IoT ecosystem is hard because of its specific features, such as limited resource, particular protocol stacks, and network requirements. For these reasons, an innovative hybrid IDS model has been proposed in this paper integrating SIDS and AIDS that can provide robust intrusion detection. Hybrid IDS is developed to counter the drawback of SIDS and AIDS, as it uses SIDS and AIDS to identify both zero-day and known attacks. The objective of the hybrid IDS is to overcome the limitations of the SIDS techniques and take advantage of the processing cost of the AIDS techniques. Therefore, HIDS has no negative impact on the node's energy consumption. However, current IDSs are not adequate to detect various attacks against the IoT systems, and they require high consumption of memory and processing. In our approach, AIDS is utilized to distinguish zero-day attacks, while SIDS is utilized to recognize known attacks. The key idea of our approach is to consolidate the benefits of both SIDS and AIDS to create robust IDS. The technique for creating and joining a few classifiers to achieve high accuracy is called boosting. SIDS is developed based on the C5.0 Decision tree classifier. Decision Trees are considered one of the most popular classification techniques. The decision tree is made up of nodes that shape a rooted tree, meaning it is a directed tree with a node called a "root" that has no incoming edges. The C5.0 decision trees provide outputs, using one attribute at a time to distinguish the data. New data can be categorized by sets of criteria defined at the nodes [8].

AIDS is developed based on a one-class Support Vector Machine (SVM). AIDS uses the known attack information and builds the profiles of normal behaviors of operations correctly. Our model contains the feature selection component for selecting suitable features, which can efficiently decrease the redundant and inappropriate features. Feature selection often leads to increased detection accuracy, reduced false alarm rate and reduced storage and computational capacity of IoT.

The main contributions of this paper are as follows:

- Development of feature selection technique based on information gain principle to select IoT features that result in maximum difference of features amongst all the applications profiled.
- Development of Hybrid Intrusion Detection System (HIDS) for IoT devices and gateways that uses a C5 classifier in the first stage and one class SVM in the second stage to create an effective ensemble architecture for improved accuracy. The experimental results show that the HIDS attains 99.97% accuracy of detection.

This paper is structured as follows. The background is provided in Section 2. Related work is discussed in Section 3. We present our approach to building models for the study in Section 4. The experimental setup is presented in Section 5. Lastly, the conclusion is presented in Section 6.

2. Background

IoT is made up of smart devices that interconnect with one another. It permits the smart devices to gather and share information. IoT devices use a back-end cloud services for intensive processing to maintain remote control [9]. Clients are able to gain access to this data and control their devices through a mobile application or web-based interface. With a large number of sensors and actuators connected to the Internet, it is important to gather raw data and apply data mining techniques to extract more interesting information about the devices to develop efficient IDSs.

Smart devices can be connected via a wired or wireless connection. The wireless connections pose security challenges, as many diverse wireless communication methods and protocols could be applied to interconnect IoT devices. These technologies include Low power Wireless Personal Area Networks (6LoWPAN), ZigBee, Bluetooth Low Energy (BLE), Z-Wave, and Near Field Communication (NFC) [10].

Figure 1 shows the IoT system architecture with layers where attacks can occur. An IoT system can comprise three fundamental layers which are the perception layer, network layer, and application layer [11]. The perception layer is the lowest layer of the conventional architecture of IoT. This layer consists of devices, sensors, actuators, and controllers. This layer's fundamental task is to gather valuable information from IoT sensors systems. Network layer ensures the successful transmission of data while application layer is the highest layer that processes the data for visualization. This layer consists of various applications that essentially use the data provided by the underlying layers.

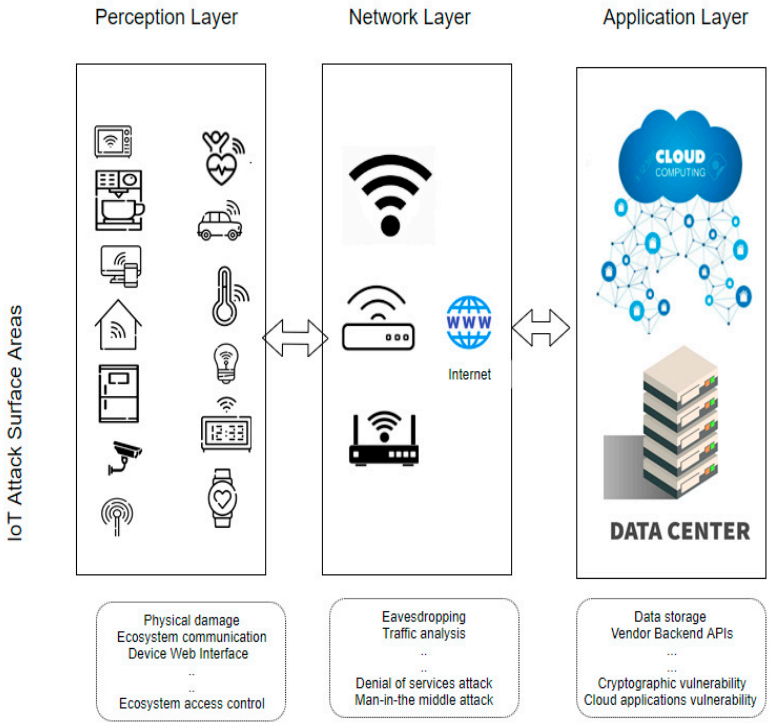

Figure 1. Internet of Things (IoT) architecture and layer attacks.

The data transfers among these levels takes place via following transmission channels:

- Device to device (D2D): peer to peer communication between two devices while using communication technologies such as Bluetooth, ZigBee, and Wi-Fi are common in the IoT system.
- Device to gateway: the gateway acts as a connection between the cloud and another node in IoT (e.g., controllers, sensors, and intelligent devices). All information to the data system is routed through the interconnected gateways. They have two main tasks: (i) to combine data from sensors and route it to the relevant data system; and (ii) to analyze data and, if a fault is detected, to initiate the recovery mechanism as per application's security requirements.
- Gateway to data systems: data sent from a gateway to a suitable data system.
- Between data systems: information transmission within data centers or clouds.

IoT Threat Model

The rapid increase of the IoT adoption also increases the number of security threats that cybersecurity researchers must consider in order to devise a robust IDS. Several types of malicious activities try to attack the security and privacy of the IoT devices and potentially all smart devices on

the publicly accessible Internet can be a target. The IoT is vulnerable against attacks for a number of reasons. Firstly, IoT devices are often unattended (e.g., sensors positioned in remote locations) and in this way, this makes it very easy for an attacker to gain access to them physically. Secondly, the greater part of the data communication is wireless, which makes it easier to eavesdrop. Lastly, the majority of the IoT devices have low storage capacities and limited processing capability. For example, additional security software could not be installed in the IoT devices.

Cybercriminals can interrupt or modify the behavior of smart devices using various hacking techniques [12]. Some of the hacking techniques need physical access to smart devices, making an attack harder to achieve, although not impossible given the physically unsecured nature of many IoT devices. Other attacks could be completed over the network from a remote site. Table 1 shows common attack types on attack smart devices.

Table 1. Common attack types which are used to attack smart devices.

IoT attack types	Description	Examples	References
Device attack	Defined as an attack in which someone takes advantage of any bug or vulnerability to gain access to the IoT infrastructure.	For smart IoT devices, such as security surveillance cameras, a cybercriminal could basically get physical access to the device and this will permit a cybercriminal to modify the design settings.	[13]
Attacks on Wi-Fi/Ethernet	Numerous malicious activities can be performed on smart IoT devices if an attacker gains physical access to the local network wirelessly.	In the network level attacks, cybercriminals are able to redirect network traffic, for example, Address Resolution Protocol poisoning (ARP) or by changing the Domain Name System (DNS) settings.	[13]
Cloud infrastructure attacks	IoT device interconnects with back-end cloud services. IoT cloud services might permit the client to select simple passwords.	A lot of cloud services have a logical weakness, which is actually the permission of cloud to a cybercriminal of obtaining sensitive information of the customer and also the access to the device without any authentication. Common vulnerabilities of management console are also contained in these services.	[14]
A Man-in-the-Middle attack (MitM)	Man-in-the-middle is a type of eavesdropping attack. This attack could permit the attacker secretly relays and possibly alters the communications between two IOT devices.	Attackers have used a network packet analyzer, i.e., Wireshark for analyzing network traffic. IoT device communicates with other IoT devices. This connectivity is not encrypted or even authenticated. That is why it is very easy for an attacker to target access to the network, thereby allowing them mount attack such as Address Resolution Protocol (ARP) poisoning.	[13]
Reconnaissance	The aim to find data about an IoT infrastructure, including the network services and devices that are running	This can be achieved by scanning network ports and packet sniffers	[15]
Connected Device—Denial of Service (DoS)	Electronic devices and its connected devices are deactivated or changed by a cybercriminal, via physical or remote access to the IoT sensors.	An attacker can deny the sensors the capability to send and receive communication. Another example could be battery abuse, device disabling, or device bricking.	[16]
Server-side Denial of Service (DoS)	Server-side functionality, set to assist smart-devices, is affected and blocked by an attacker, attacking the sensor from his own smart-device.	DoS can flood devices with overwhelming traffic.	[17]
IoT Botnet	Group of hacked computers, smart devices, and appliances connected to the Internet are known as IoT botnet, these devices are the one chosen for attacks. They mainly attack online clients and devices such as IP cameras and home routers.	The Mirai malware is seen as a milestone in the threat landscape and exploits security holes in IoT devices and launches attacks [18].	[19]
Privilege escalation	The attacker takes advantage of programming errors or software flaws to permit cybercriminals to elevate access to IoT infrastructure.	Grant the cybercriminal elevated access to the IoT ecosystem and its associated data and applications.	[16]

An IoT botnet consisting of exposed IoT devices, such as electronic appliances, security systems, cars, thermostats and lights in private or commercial environments, speaker systems, alarm clocks, vending machines, and many other can be affected by the intrusion attacks. These intrusions permit a cybercriminal to control the sensors. Dissimilar to conventional botnets, affected IoT devices search to spread their malicious activity to an ever-increasing number of devices. A conventional botnet may comprise thousands of bots, but IoT botnet is bigger in scale, with a large number of attached devices [20]. For example, a large DNS server company called Dyn was targeted by cyber attackers on October 21, 2016. This attack was actually launched by an extraordinarily large number of DNS lookup requests from tens of millions of IP addresses [21]. The requests from the Botnet infected a large number of internet connected devices like printers, digital cameras, and other devices. This IoT botnet attack was caused by malicious software named Mirai. Due to Mirai infection, computers persistently browse the internet for devices that are vulnerable and use default username and password to access the system, infecting them with malicious software.

At Black Hat 2015, security researchers revealed how they attacked Chrysler Jeep Cherokee. While attacking the Jeep's system of IoT devices and sensors, one could remotely control a Jeep as it drives down the highway [22].

3. Related Work

In this section, a review of the existing IDS research for IoT is presented. Each research was categorized by considering the following characteristics: IDS placement strategy, detection method, and validation strategy. Figure 2 shows the classification of IDS for IoT networks, while Table 2 provides some recent related research.

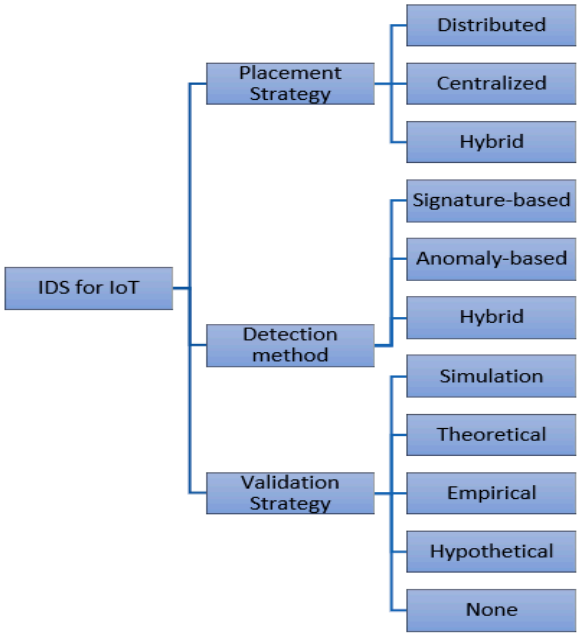

Figure 2. Classification of Intrusion Detection Systems for IoT.

In IDS placement strategies, IDS can be classified as distributed, centralized, or hybrid. In distributed placement, the IoT devices could be responsible for checking other IoT devices.

In the centralized IDS location, the IDS is placed in central devices, for instance, in the boundary switch or a nominated device. All the information that the IoT devices collect and then send to the network boundary switch passes through the boundary switch. Consequently, the IDS positioned in a boundary switch can check the packets switched between the IoT devices and the network. In spite of this, checking the net-work packets that pass through the boundary switch is not adequate to identify anomalies that affect the IoT devices.

Raza et al. used a hybrid, centralized, and distributed approach and placed IDS modules both in the border router and in the nominated nodes [23]. They applied signature- and anomaly-based techniques to detect routing attacks, where an attacker provides nearby nodes with false routing data and then modifies the data that transmit through it.

Current works on IDS for IoT have three primary classes: Anomaly-based Intrusion Detection System (AIDS), Signature-based Intrusion Detection Systems (SIDS), and hybrid. In short, SIDS relies on pattern matching techniques for finding known attacks; these are also known as Knowledge-based Detection or Misuse Detection [24]. In SIDS, matching methods are used to find a previous intrusion. In AIDS, a normal model of the behavior of a computer system is determined using machine learning, statistical-based or knowledge-based methods. Any significant deviation between the observed behavior and the model is regarded as an anomaly, which can be interpreted as an intrusion. The assumption for this group of techniques is that malicious behavior differs from typical user behavior, while the Hybrid IDS methodology combines SIDS with AIDS to improve the detection rate and decrease false alarms.

To validate the effectiveness of IDSs, researchers have used different techniques, such as theoretical, empirical, and hypothetical strategies, for validating their techniques.

Hoda et al. used AIDS based on a neural network for detecting Denial of Service attacks over the IoT networks. Their IDS approach was based on categorizing normal and abnormal patterns. The AIDS model was tested against a simulated IoT network [25].

Diro et al. developed an IoT network attack detection system on the basis of distributed deep learning. Their work showed that distributed attack detection could identify IoT attacks better than a centralized strategy with 96% detection rate. Their approach was evaluated using NLS-KDD dataset. Even though this dataset is another version of the KDD data set, it still suffers from various issues reviewed by McHugh [26]. We believe this dataset should not be used as an effective bench-mark dataset in the IoT, as this data was collected from the traditional network [27]. This leads us to develop IDSs that take in consideration the specific requirement of IoT protocol such as Low-power Wireless Personal Area Networks (6LowPAN). Hence, intrusion detection system that is created for the IoT ecosystem should operate under rigorous settings of low processing ability, high speed connection, and big capacity data processing.

Table 2. Summary of the proposed research to IDSs for IoT.

Key References	Placement Strategy	Detection Techniques	Security Threat	Validation Strategy
Cho et al. [29]	Centralized	AIDS	Botnet	Simulation
Raza et al. [23]	Hybrid, centralized and distributed	Hybrid	Routing attacks	Simulation
Rathore and Park [28]	Distributed	AIDS	Network attack	Empirical (NSL-KDD Dataset)
Hodo et al. [25]	Centralized	AIDS	DoS attack	Simulation
Diro and Chilamkurti [27]	Distributed	AIDS	Network attack	Empirical (NSL-KDD Dataset)
Moustafa et al. [30]	Distributed	Hybrid	The botnet, Man in the Middle	Empirical
Cervantes et al. [31]	Distributed	Hybrid	Sinkhole attacks	Simulation
Venkatraman and Surendiran [32]	Distributed	Hybrid	DoS, control hijacking and replay	Simulation

Rathore et al. proposed semi-supervised Fuzzy learning based distributed attack detection framework for IoT [28]. The evaluation was done on the Network Security Laboratory - Knowledge Discovery in Databases (NSL-KDD) dataset and consequently suffers from the same dataset limitations as mentioned above.

Cho et al. proposed a methodology for checking packets that are passing through the border router for communication between physical and the network devices. Their methodology was based on the botnet attacks by checking the packet length [29]. However, no information is presented about

the technique can be employed to create normal behavior profiles. It is also not clear how the proposed IDS techniques would work on resource constraints nodes in the IoT.

Moustafa et al. proposed an ensemble of IDSs to detect abnormal activities, in specific botnet attacks against Domain Name System (DNS), Hyper Text Transfer Protocol (HTTP), and Message Queue Telemetry Transport (MQTT) [30]. Their ensemble methods are based on the AdaBoost learning method and they used three machine learning techniques: Artificial Neural Networks (ANN), Decision Tree (DT), and Naive Bayes (NB) to evaluate their methodology [30]. The proposed IDS results in significant overhead, which degrades its performance.

Hodo et al. used an Artificial Neural Network (ANN) to detect DDoS and DoS attacks against legitimate IoT network traffic. The proposed ANN model was tested with the use of a simulated IoT network. Hoda et al. proposed a threat analysis of IoT using ANN to detect DDoS/DoS attacks. A multi-level perceptron, a type of supervised ANN, was trained using internet packet traces, and then, the model was assessed on its ability to thwart (DDoS/DoS) attacks [25]. Hoda et al. did not consider the effectiveness after the deployment of the proposed IDS in the IoT ecosystem on low capacity devices. According to their experimentation, the system achieved an accuracy of 99.4% for DDoS/DoS. However, no details of the dataset are provided.

Cervantes et al. proposed IDS for detecting sinkhole attacks on 6LoWPAN for the IoT. Their IDS approach applies a combination of anomaly detection and support vector machine (SVM). Each IDS agent trains the SVM, and executes a majority voting decision to mark the infected nodes [31]. Their simulation results showed that their IDS achieve a sinkhole detection rate up to 92% on the fixed scenario and 75% in a mobile scenario. However, their approach has not been evaluated for other types of attacks in the IoT.

Patil and Modi [33] designed a virtual environment monitoring system to prevent intrusions in IoT. This system used predefined signature database for known attacks and it applies anomaly-based detection for unknown attacks.

Table 3 shows the IDS techniques and datasets covered by this paper and previous research papers.

Table 3. Comparison of this research and similar researches: (✔: Topic is covered, ✖ the topic is not covered).

Related Researches	Intrusion Detection System Techniques						IoT Dataset
	SIDS	AIDS				Hybrid IDS	
		Supervised Learning	Unsupervised	Semi-Supervised Learning	Ensemble Methods		
Kenkre et al. [34]	✔	✖	✖	✖	✖	✖	✖
Hodo et al. [25]	✖	✔	✖	✖	✖	✖	✔
Liao et al. [35]	✔	✔	✔	✖	✖	✔	✖
Ashfaq et al. [36]	✖	✖	✖	✔	✖	✖	✖
Al-Yaseen et al. [37]	✖	✖	✖	✖	✖	✔	✖
This paper	✔	✔	✖	✖	✔	✔	✔

4. Proposed Hybrid Model for IDS

Hybrid IDS has been proposed to overcome the shortcomings of SIDS and AIDS, as it brings together SIDS and AIDS to identify both unknown and known attacks. Novel techniques were used to combine the results of SIDS and AIDS. In our methodology, AIDS was utilized to recognize zero-day attacks, while SIDS was utilized to distinguish well-known attacks. Boosting method was used to combine the classifiers and to decrease the bias of the combined model. The Hybrid IDS has two stages; the SIDS stage and AIDS stage, as shown in Figure 3. AIDS aims to profile the normal nodes activity and would raise a malicious alarm when the difference between normal requests exceeds the predefined threshold for a given observation. Nodes' profiles were created by employing records that

were recognized as benign actions. Next, it observed the behavior of the traffic and matches the new records with the built profiles and attempts to identify abnormalities. If any malicious request was identified, the system will save it in the signature database. The main purpose of storing the malicious pattern in the database was to achieve protection against the similar attacks in upcoming malicious activity. In other words, the SIDS will have an appropriate history of previously known attacks.

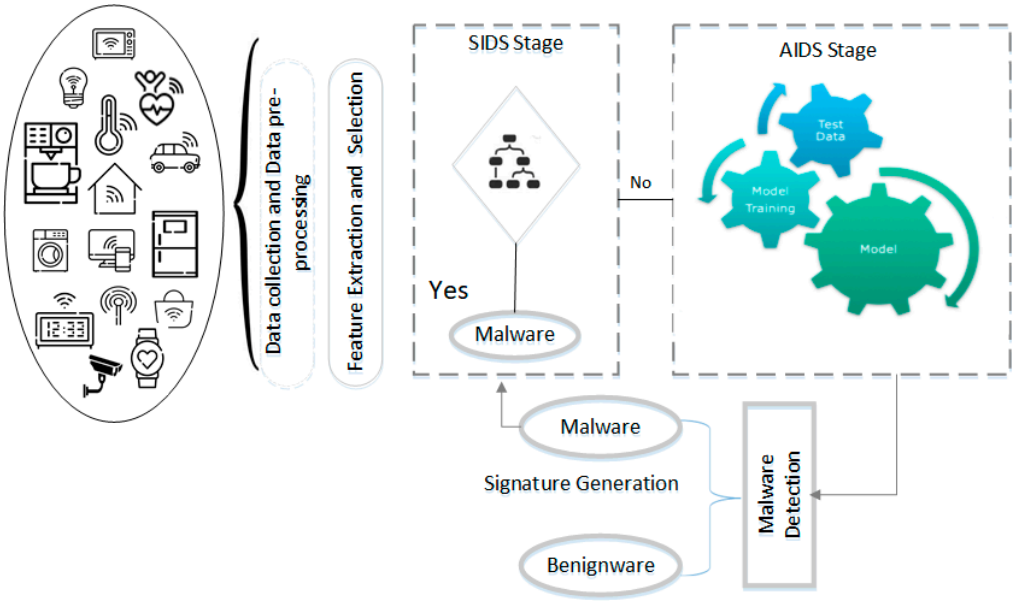

Figure 3. Hybrid Intrusion Detection System for the IoT ecosystem.

4.1. Feature Selection

The IoT ecosystem is made up of smart devices with limited processing power, memory, energy, and communication range. One main issue among many others with IDSs is dealing with many irrelevant features, which can cause overhead on the system. It is well known that redundant, irrelevant features often lead to low detection rate. Therefore, the purpose of the feature selection is to identify significant features which can be used in the IDS to detect various attacks efficiently [38].

With the extracted labels, the features are analyzed for both normal and abnormal behaviors to determine the most relevant features. We applied an information gain method for feature selection. The information gain methods had a fast execution time and this technique extracted the best performing feature set for the particular type of model. In literature, information gain was regularly applied to assess how well each distinct attribute separates the given data set. The overall entropy "I" of a given dataset "S" is described as [39]:

$$I(S) = -\sum_{i=1}^{c} p_i log_2 p_i \quad (1)$$

where, "C" refers to the total number of classes and p_i denotes the portion of instances that belongs to class i. The decrease in entropy or information gain is calculated for every feature according to:

$$IG(S,A) = I(S) - \sum_{v \varepsilon A} \frac{|S_{A,v}|}{|S|} I(S_v) \quad (2)$$

where v values of A and $S_{A,v}$ are the instances of a set.

4.2. Building Classification Models

Once the selected features are identified, we ran experiments using Hybrid IDS to evaluate their capability to distinguish malicious activities from normal activities. Our Hybrid IDS model involved two phases, namely SIDS and AIDS.

4.3. Stage One: SIDS Stage

In the SIDS phase, C5 decision tree classifier was used to create a decision tree [40]. Once a decision tree is created, it can be applied to detect other samples with varying success depending on how well it models the dataset. The tree can then be applied as a rule set for detecting whether a test sample is malware or benign software.

Unknown traffic was handled through pattern matching to determine whether it represents normal or abnormal activities. If the request matches with an attack signature from the database, it raises an alarm that it is a malicious sample. If it did not match, it will go to AIDS, which is the next stage of the framework as shown in Figure 3, as AIDS is designed to detect unknown attacks, such as a zero-day attack.

4.4. Stage Two: AIDS Stage

In order to effectively recognize unknown attacks, the output of SIDS-stage is used to train AIDS to recognize abnormal activities. AIDS, being trained using benign samples, should be able to separate activities which do not appear to be normal, i.e., unusual behaviors exhibited by malware type software. To train AIDS, One-Class SVM is used, which learns the attributes of benign samples without using any information from the other class. Such a classifier can identify normal activities with far more success as normal class training data are easily available. In contrast, zero-day attacks are rare. Hence, we may have few instances of training datasets for zero-day attacks or even none.

Therefore, in the second stage, normal behavior is identified, and anything outside the normal behavior is classified as a zero-day attack. One-class classification techniques aim to build classification models when the malware class is unavailable, poorly sampled, or not well identified. The unique circumstances constrain the learning of efficient classifiers by describing class boundary just with the information of the normal class. In contrast to the traditional multi-class classification paradigm, in one-class classification, normal behavior is well described by examples in the training data, while the unknown malware has no example.

4.5. Stage Three: Stacking of the Two Stages

SIDS and AIDS have correlative qualities and shortcomings; thus, we propose to build up a hybrid method utilizing an ensemble of both approaches. In machine learning, ensemble techniques are used to enhance prediction accuracy. Although many ensemble methods have been proposed, this is a difficult task to find an appropriate ensemble configuration for detecting the zero-day attack. A C5 classifier was used as a first stage and a one class SVM was used as the second stage to create an ensemble of classifiers to improve accuracy for IDS.

5. Experimental Setup

The Bot-IoT dataset is used to evaluate the proposed hybrid IDS. The experiments have been performed using C5 and LIBSVM with default parameters. Details of the datasets are presented below.

5.1. Dataset

The Bot-IoT dataset, which includes normal IoT network traffic along with a variety of attacks, was used to evaluate our proposed framework. This dataset was selected because it represents a realistic IoT ecosystem environment. The dataset contains DDoS, DoS, OS and Service Scan, Keylogging, and Data exfiltration attacks. All these data were pre-processed to identify network-level patterns for

diverse kinds of traffic that devices create; and use these patterns to detect any intrusion behaviors in the IoT Infrastructure [20]. The features and their descriptions are presented in Table 4 while the number of benign and attack samples in the dataset is shown in Figure 4.

Table 4. Features for IoT networks.

Feature	Description
pkSeqID	Row Identifier
Stime	Record start time
Flags	Flow state flags were seen in transactions
flgs_number	Numerical representation of feature flags
Proto	A textual representation of transaction protocols presents in network flow
proto_number	Numerical representation of feature proto
Saddr	Source IP address
Sport	Source port number
Daddr	Destination IP address
Dport	Destination port number
Pkts	Total count of packets in a transaction
Bytes	Total number of bytes in the transaction
State	Transaction state
state_number	Numerical representation of feature state
Ltime	Record last time
Seq	Argus sequence number
Dur	Record total duration
Mean	The average duration of aggregated records
Stddev	The standard deviation of aggregated records
Sum	The total duration of aggregated records
Min	The minimum duration of aggregated records
Max	The maximum duration of aggregated records
Spkts	Source-to-destination packet count
Dpkts	Destination-to-source packet count
Sbytes	Source-to-destination byte count
Dbytes	Destination-to-source byte count
Rate	Total packets per second in transaction
Srate	Source-to-destination packets per second
Drate	Destination-to-source packets per second
TnBPSrcIP	Total Number of bytes per source IP
TnBPDstIP	Total Number of bytes per Destination IP
TnP_PSrcIP	Total Number of packets per source IP
TnP_PDstIP	Total Number of packets per Destination IP
TnP_PerProto	Total Number of packets per protocol
TnP_Per_Dport	Total Number of packets per dport
AR_P_Proto_P_SrcIP	Average rate per protocol per Source IP (calculated by pkts/dur)
AR_P_Proto_P_DstIP	Average rate per protocol per Destination IP
N_IN_Conn_P_SrcIP	A number of inbound connections per source IP
N_IN_Conn_P_DstIP	Number of inbound connections per destination IP
AR_P_Proto_P_Sport	Average rate per protocol per sport
AR_P_Proto_P_Dport	Average rate per protocol per sport
Pkts_P_State_P_Protocol_P_DestIP	A number of packets grouped by the state of flows and protocols per destination IP
Pkts_P_State_P_Protocol_P_SrcIP	A number of packets grouped by the state of flows and protocols per source IP
Attack	Class label: 0 for Normal traffic, 1 for Attack Traffic
Category	Traffic category
Subcategory	Traffic subcategory

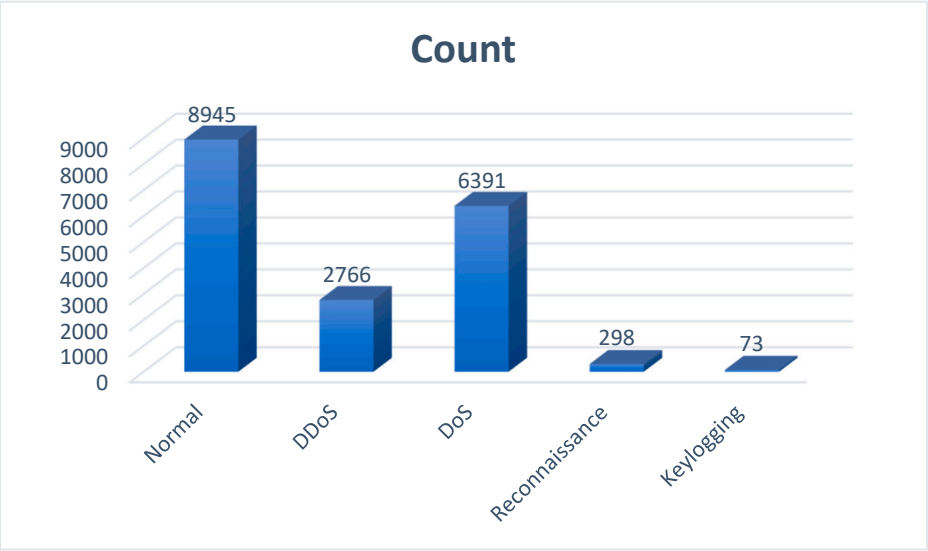

Figure 4. Statistics of attacks and normal behavior in the Bot-IoT dataset.

5.2. Evaluation Metrics for Models

In this experiment, we have evaluated the effectiveness of our IDS with the use of the Confusion Matrix as shown in Table 5. The following metrics were calculated:

Table 5. Confusion matrix.

	Predicted Attack	**Predicated Normal**
Actual Attack	True positive (TP)	False Negative
Actual Normal	False Positive	True Negative

True positive (TP): the number of rightly recognized malicious code.

True negative (TN): the number of rightly recognized benign code.

False positive (FP): the number of incorrectly identified benign code, when a detector recognizes a benign code as a Malware.

False negative (FN): the number of incorrectly recognized malicious code, when a detector recognizes a Malware as a benign code.

Total Accuracy: proportion of accurately classified instances, either positive or negative. The accuracy is calculated according to the following equation:

$$Accuracy = (TP + TN)/(TP + FN + FP + TN) \quad (3)$$

False alarm rate of IDSs can be computed by $F(i) = 1 - \frac{Xii}{\sum_{i=1}^{6} Xij}$

Table 5 shows the confusion matrix for two classes and Table 6 shows the confusion matrix for six classes, one normal and five attacks, the element X $(1 <= i <= 6;\ 1 <= j <= 6)$ denotes the number of records that belong to class i and were classified as class j by the IDS. On the basis of the confusion matrix, one can easily compute performance criteria, for example, the detection rate of class i: $DR(i) = \frac{Xii}{\sum_{i=1}^{6} Xij}$.

Table 6. Confusion matrix to evaluate the performance of our proposed IDS.

Classified as	a	b	c	d	E	f
a = Normal	X11	X12	X13	X14	X15	X16
b = DDoS	X21	X22	X23	X24	X25	X26
c = DoS	X31	X32	X33	X34	X35	X36
d = Reconnaissance	X41	X42	X43	X44	X45	X46
e = Keylogging	X51	X52	X53	X54	X55	X56

5.3. Experimental Results

In this section, we provide the detailed results of the experiments using the proposed framework. The proposed model is applied to the IoT ecosystem, and its performance is evaluated against the other state-of-the-art machine learning techniques using the BoT-IoT intrusion dataset. To evaluate the system's accuracy for all stages, four measures were computed: true positive rate, F-measure, false positive rate, and accuracy.

Feature Selection Results

For both benign and malicious classes, information gain is calculated, and we removed the feature sets whose information gain was less than predetermined thresholds (set arbitrarily to 0.2). This calculation involves the estimation of the conditional probabilities of a class for a given feature, and entropy computations. The feature with good information gain is considered the most discriminative feature. For example, if the ranked value is higher than threshold then it indicates that a feature is useful for distinguishing this class, Otherwise, it will be eliminated from the feature sets. To get a better threshold value, the distribution of the Information Gain (IG) values is computed and verified with diverse threshold numbers on the training dataset.

$$IG(t) = -\sum_{i=1}^{m} P(c_i) log\ _{P(c_i)} + p(t) \sum_{i=1}^{m} p(c_i|t) log\ p(c_i|t) + P(\acute{t}) \sum_{i=1}^{m} P(c_i|\acute{t}) log\ \acute{P}(c_i|\acute{t}) \quad (4)$$

where:

- c_i represents *(i)* category.
- $P(c_i)$: probability that random instance belongs to class c_i
- *P(t)* and $P(\overrightarrow{t})$*:* probability of the occurrence of the feature w in a randomly selected document.
- $P(c_i|t)$: probability that a randomly selected instance belongs to class c_i. if instance has the feature w.
- m is the number of classes.

Table 7 presents the information gain of IoT features. In total, 43 features are examined. Among them, 13 features are the most significant concerning malware detection. A higher rank of a feature makes it suitable to distinguish well between normal and malware applications. The features in Table 7 are presented in the descending order of their contribution in identifying malware.

Table 7. Information gain for different features.

Ratio	Feature Name
0.7579	dport
0.6433	seq
0.62	dur
0.4393	flgs_number
0.4381	flgs
0.3547	sport
0.3346	N_IN_Conn_P_DstIP
0.3023	srate
0.2873	AR_P_Proto_P_Sport
0.2817	daddr
0.2788	TnBPDstIP
0.2772	rate
0.274	AR_P_Proto_P_SrcIP

Stage one: SIDS Results:

We have used the k–fold cross-validation technique for performance evaluation. In this technique, the dataset is randomly divided into k different parts. For each iteration, one-fold was selected for testing and all other (k-1) folds were treated as the training dataset. For all experiments, the value of k was taken as 10 because of low bias; low variance, low overfitting, and good error estimation. The folds were stratified so that the class was characterized in approximately the same proportions as in the full dataset. Each fold was held out one by one and the learning scheme was trained on the remaining nine folds; then its error rate was calculated for the holdout set. The learning procedure was performed 10 times on different training sets, and finally, the 10 error rates were averaged to yield an overall error estimate.

To assess the performance of the proposed technique, the confusion matrix is used. Confusion matrix results for C5 classifier in stage one is shown in Table 8. The detailed analysis of the accuracy of C5 decision tree classification is shown in Table 9. Detailed accuracy of C5 decision tree.

Table 8. Confusion Matrix results of using C5 classifier.

Classified as	a	b	c	d	E
a = Normal	7728	0	0	1217	0
b = DDoS	0	2754	12	0	0
c = DoS	0	0	6384	7	0
d = Reconnaissance	0	0	1	297	0
e = Keylogging	0	0	0	0	73

Table 9. Detailed accuracy of C5 decision tree.

Class	TP Rate	FP Rate	F-Measure
Normal	0.864	0	0.927
DDoS	0.996	0	0.998
DoS	0.999	0.001	0.998
Reconnaissance	0.997	0.067	0.327
Keylogging	1	0	1
Weighted Avg	0.933	0.001	0.953

Stage two: AIDs Results:

One-class SVM with RBF kernel was implemented using LIBSVM. Results in the form of a confusion matrix of stage two are shown in Table 10.

Table 10. Confusion matrix of using One-Class Support Vector Machine.

Classified as	a	b
a = Normal	7618	1327
b = Intrusion	4	9524

The detailed analysis of the accuracy of One-Class SVM classifier result is shown in Table 11.

Table 11. Detailed accuracy of using one class SVM.

Class	TP Rate	FP Rate	F-Measure
Normal	0.852	0	0.920
Intrusion	1	0.148	0.935
Weighted Avg	0.928	0.077	0.927

Stage Three: The Combination of the two stages:

In Hybrid IDS, the C5 classifier is applied as a first stage, and one class SVM is employed in the second stage to develop hybrid IDS. Stacking ensemble method is used to combine the two stages. Confusion matrices of the combination of the classifiers in stage three is shown in Table 12. The details accuracy of stage 3 is shown in Table 13.

Table 12. Confusion matrix with the use of Hybrid classification.

Classified As	a	B	c	D	e
a = Normal	7869	0	0	1076	0
b = DDoS	0	2737	29	0	0
c = DoS	0	0	6384	7	1
d = Reconnaissance	0	0	2	296	0
e = Keylogging	0	0	0	0	73

Table 13. Detailed accuracy of using stage 3.

Class	TP Rate	FP Rate	F-Measure
Normal	0.88	0	0.936
DDoS	0.99	0	0.995
DoS	0.999	0.003	0.997
Reconnaissance	0.993	0.06	0.353
Keylogging	1	0	1
Weighted Avg	0.94	0.002	0.957

As shown in Figure 5, the accuracy of detection of malware is 94% on the IoT intrusion dataset in stage one, while it is 92.5 % in stage two. In stage 3, the accuracy results have been improved to 99.97%. Therefore, the proposed framework yields higher detection accuracy and lower false alarm rate in contrast to the standalone single stage.

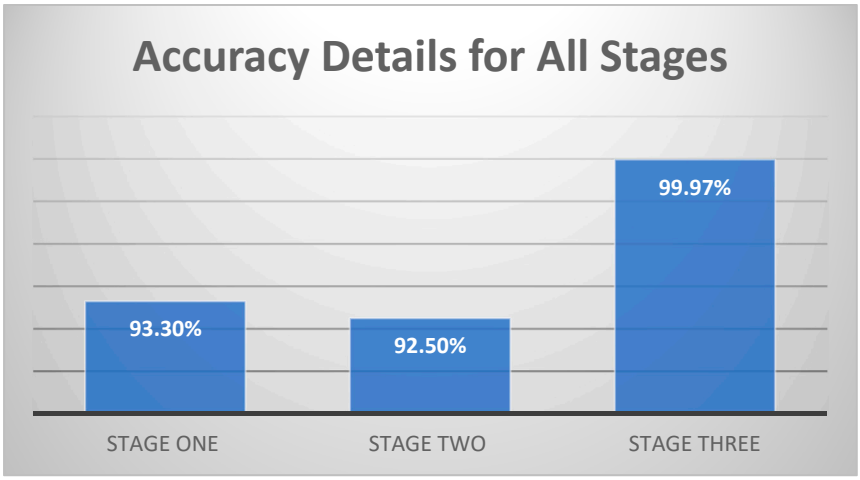

Figure 5. Accuracy details for all stages.

Table 14 shows the performance of different machine learning techniques, namely C4.5, Naïve Bayes, Random Forest, multi-layer perception, Support Vector Machine (SVM), Classification and Regression Tree (CART), and K Nearest Neighbor (KNN) on the Bot-IoT dataset. The results show that the combination of the two stages provides the best performance attaining an accuracy of 99.97%.

Table 14. The performance of different machine learning techniques.

Machine Learning Techniques	Accuracy
C4.5 [8]	92%
Naïve Bayes	87.56%
Random Forest	92.67%
Multi-layer perception	87.41%
SVM	89.52%
CART	80.3%
KNN	88.4%
Proposed Technique	99.97%

6. Conclusions

This paper presents the design, implementation, and evaluation of proposed novel IDS for intrusion detecting for IoT infrastructure. The proposed system relies on the feature set extracted from IoT ecosystem to effectively detect various types of IoT attacks. A set of features is used to create an effective Hybrid IDS for detecting IoT attacks. Experimental results show that combining two stages of the proposed framework through stacking ensemble method improves the detection accuracy. We have shown that an ensemble of C5 and one-class SVM in two cascaded stages is superior to individual techniques. Our experimental results show that our suggested hybrid IDS has superior performance overall in terms of accuracy and false alarm rate compared with the other machine learning techniques and approaches reported in previous studies. This suggests that our proposed technique will be very useful in designing modern IDSs. Future work includes extending the proposed IDS to detect other types of attacks against IoT systems.

Author Contributions: A.K. is the main author of the current paper. A.K. contributed to the development of the ideas, design of the study, theory, result analysis, and article writing. A.K. also designed the experiments and then performed the experiments. I.G., P.V., J.K., and A.A. undertook the revision works of the paper. All authors read and approved the final manuscript.

Funding: This work is done in Internet Commerce Security Lab, which is funded by Westpac Banking Corporation.

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

Abbreviations

IDSs	Intrusion Detection Systems
SIDS	Signature Intrusion Detection
AIDS	Anomaly Intrusion Detection System
AI	Artificial Intelligence
CPU	Central Process Unit
FN	False Negative
FP	False Positive
HIDS	Host-based Intrusion Detection System
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
HIDS	Hybrid Intrusion Detection System
IoT	Internet of Things

References

1. Sarkar, S.; Chatterjee, S.; Misra, S. Assessment of the Suitability of Fog Computing in the Context of Internet of Things. *IEEE Trans. Cloud Comput.* **2018**, *6*, 46–59. [CrossRef]
2. Chowdhury, A.; Karmakar, G.; Kamruzzaman, J. The Co-Evolution of Cloud and IoT Applications: Recent and Future Trends. In *Handbook of Research on the IoT, Cloud Computing, and Wireless Network Optimization*; IGI Global: Hershey, PA, USA, 2019; pp. 213–234.
3. Saha, H.N.; Mandal, A.; Sinha, A. Recent trends in the Internet of Things. In Proceedings of the 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 9–11 January 2017; pp. 1–4.
4. Yar, M.; Steinmetz, K.F. *Cybercrime and Society*; SAGE Publications Limited: Thousand Oaks, CA, USA, 2019.
5. Symantec. Internet Security Threat Report. Available online: https://www.symantec.com/security-center/threat-report (accessed on 23 Feburary 2018).
6. Khraisat, A.; Gondal, I.; Vamplew, P.; Kamruzzaman, J. Survey of intrusion detection systems: Techniques, datasets and challenges. *Cybersecurity* **2019**, *2*, 20. [CrossRef]
7. Huda, S.; Abawajy, J.; Alazab, M.; Abdollalihian, M.; Islam, R.; Yearwood, J. Hybrids of support vector machine wrapper and filter based framework for malware detection. *Future Gener. Comp. Sy.* **2016**, *55*, 376–390. [CrossRef]
8. Mienye, I.D.; Sun, Y.; Wang, Z. Prediction performance of improved decision tree-based algorithms: a review. *Procedia Manuf.* **2019**, *35*, 698–703. [CrossRef]
9. Zarpelao, 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]
10. Al-Sarawi, S.; Anbar, M.; Alieyan, K.; Alzubaidi, M. Internet of Things (IoT) communication protocols: Review. In Proceedings of the 2017 8th International Conference on Information Technology (ICIT), Amman, Jordan, 17 May 2017; pp. 685–690.
11. Sonar, K.; Upadhyay, H. A survey: DDOS attack on Internet of Things. *Int. J. Eng. Res. Dev.* **2014**, *10*, 58–63.
12. Alazab, A.; Abawajy, J.; Hobbs, M.; Layton, R.; Khraisat, A. Crime toolkits: the productisation of cybercrime. In Proceedings of the 2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, Melbourne, Australia, 16–18 July 2013; pp. 1626–1632.
13. Barcena, M.B.; Wueest, C. *Insecurity in the Internet of Things*; Symantec: Mountain View, CA, USA, 2015.
14. Singh, J.; Pasquier, T.; Bacon, J.; Ko, H.; Eyers, D. Twenty Security Considerations for Cloud-Supported Internet of Things. *IEEE Internet Things J.* **2016**, *3*, 269–284. [CrossRef]
15. Abomhara, M. Cyber security and the internet of things: vulnerabilities, threats, intruders and attacks. *J. Cyber Secur. Mobil.* **2015**, *4*, 65–88. [CrossRef]
16. Bertino, E.; Islam, N. Botnets and internet of things security. *Computer* **2017**, *2*, 76–79. [CrossRef]
17. Lu, Y.; da Xu, L. Internet of Things (IoT) cybersecurity research: A review of current research topics. *IEEE Internet Things J.* **2018**, *6*, 2103–2115. [CrossRef]

18. Kolias, C.; Kambourakis, G.; Stavrou, A.; Voas, J. DDoS in the IoT: Mirai and other botnets. *Computer* **2017**, *50*, 80–84. [CrossRef]
19. Sarang, R. Trending: IoT Malware Attacks of 2018. Available online: https://securingtomorrow.mcafee.com/consumer/mobile-and-iot-security/top-trending-iot-malware-attacks-of-2018/ (accessed on 3 March 2018).
20. Koroniotis, N.; Moustafa, N.; Sitnikova, E.; Turnbull, B. Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-IoT Dataset. *arXiv* **2018**, arXiv:1811.00701. [CrossRef]
21. Mansfield-Devine, S. DDoS goes mainstream: how headline-grabbing attacks could make this threat an organisation's biggest nightmare. *Netw. Secur.* **2016**, *2016*, 7–13. [CrossRef]
22. Greenberg, A. Hackers remotely kill a jeep on the highway—with me in it. *Wired* **2015**, *7*, 21.
23. Raza, S.; Wallgren, L.; Voigt, T. SVELTE: Real-time intrusion detection in the Internet of Things. *Ad Hoc Netw.* **2013**, *11*, 2661–2674. [CrossRef]
24. Alazab, A.; Hobbs, M.; Abawajy, J.; Khraisat, A.; Alazab, M. Using response action with intelligent intrusion detection and prevention system against web application malware. *Inf. Manag. Comput. Secur.* **2014**, *22*, 431–449.
25. Hodo, E.; Bellekens, X.; Hamilton, A.; Dubouilh, P.L.; Iorkyase, E.; Tachtatzis, C.; Atkinson, R. Threat analysis of IoT networks using artificial neural network intrusion detection system. In Proceedings of the 2016 International Symposium on Networks, Computers and Communications (ISNCC), Yasmine Hammamet, Tunisia, 11–13 May 2016; pp. 1–6.
26. McHugh, J. Testing Intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory. *ACM Trans. Inf. Syst. Secur.* **2000**, *3*, 262–294. [CrossRef]
27. Diro, A.A.; Chilamkurti, N. Distributed attack detection scheme using deep learning approach for Internet of Things. *Future Gener. Comput. Syst.* **2018**, *82*, 761–768. [CrossRef]
28. Rathore, S.; Park, J.H. Semi-supervised learning based distributed attack detection framework for IoT. *Appl. Soft Comput.* **2018**, *72*, 79–89. [CrossRef]
29. Cho, E.J.; Kim, J.H.; Hong, C.S. *Attack Model and Detection Scheme for Botnet on 6LoWPAN*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 515–518.
30. Moustafa, N.; Turnbull, B.; Choo, K.R. An Ensemble Intrusion Detection Technique based on proposed Statistical Flow Features for Protecting Network Traffic of Internet of Things. *IEEE Internet Things J.* **2019**, *6*, 4815–4830. [CrossRef]
31. Cervantes, C.; Poplade, D.; Nogueira, M.; Santos, A. Detection of sinkhole attacks for supporting secure routing on 6LoWPAN for Internet of Things. In Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), Ottawa, ON, Canada, 11–15 May 2015; pp. 606–611.
32. Venkatraman, S.; Surendiran, B. Adaptive hybrid intrusion detection system for crowd sourced multimedia internet of things systems. *Multimed. Tools Appl.* **2019**, 1–8. [CrossRef]
33. Patil, R.; Modi, C. *Designing a Virtual Environment Monitoring System to Prevent Intrusions in Future Internet of Things*; Springer: Singapore, 2019; pp. 345–351.
34. Kenkre, P.S.; Pai, A.; Colaco, L. Real Time Intrusion Detection and Prevention System. In Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014, Bhubaneswar, India, 14–15 November 2014; Satapathy, S.C., Biswal, B.N., Udgata, S.K., Mandal, J.K., Eds.; Springer International Publishing: Cham, Switzerland, 2015; Volume 1, pp. 405–411.
35. Liao, H.J.; Lin, C.H.R.; Lin, Y.C.; Tung, K.Y. Intrusion detection system: A comprehensive review. *J. Netw. Comput. Appl.* **2013**, *36*, 16–24. [CrossRef]
36. Ashfaq, R.A.R.; Wang, X.Z.; Huang, J.Z.; Abbas, H.; He, Y.L. Fuzziness based semi-supervised learning approach for intrusion detection system. *Inf. Sci.* **2017**, *378*, 484–497. [CrossRef]
37. Al-Yaseen, W.L.; Othman, Z.A.; Nazri, M.Z.A. Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system. *Expert Syst. Appl.* **2017**, *67*, 296–303. [CrossRef]
38. Alazab, A.; Hobbs, M.; Abawajy, J.; Alazab, M. Using feature selection for intrusion detection system. In Proceedings of the 2012 International Symposium on Communications and Information Technologies (ISCIT), Gold Coast, Australia, 2–5 October 2012; pp. 296–301.

39. Gray, R.M. *Entropy and Information Theory*; Springer: Berlin/Heidelberg, Germany, 2010.
40. Khraisat, A.; Gondal, I.; Vamplew, P. An Anomaly Intrusion Detection System Using C5 Decision Tree Classifier. In *Trends and Applications in Knowledge Discovery and Data Mining*; Springer International Publishing: Cham, Switzerland, 2018; pp. 149–155.