Boosting-Based DDoS Detection in Internet of Things Systems

Ivan Cvitić, Dragan Peraković, *Member, IEEE*, Brij B. Gupta, *Senior Member, IEEE*, and Kim-Kwang Raymond Choo, *Senior Member, IEEE*

***Abstract*—Distributed Denial-of-Service (DDoS) attacks remain challenging to mitigate in the existing systems, including in-home networks that comprise different Internet of Things (IoT) devices. In this article, we present a DDoS traffic detection model that uses a boosting method of logistic model trees for different IoT device classes. Specifically, a different version of the model will be generated and applied for each device class since the characteristics of the network traffic from each device class may have subtle variation(s). As a case study, we explain how devices in a typical smart home environment can be categorized into four different classes (and in our context, Class 1—very high level of traffic predictability, Class 2—high level of traffic predictability, Class 3—medium level of traffic predictability, and Class 4—low level of traffic predictability). Findings from our evaluations show that the accuracy of our proposed approach is between 99.92% and 99.99% for these four device classes. In other words, we demonstrate that we can use device classes to help us more effectively detect DDoS traffic.**

***Index Terms*—Artificial intelligence, cybersecurity, Distributed Denial of Service (DDoS), ensemble machine learning, IDS, Internet of Things (IoT), supervised learning.**

I. Introduction

INTERNET OF THINGS (IoT) devices and systems are becoming commonplace and, hence, they are increasingly targeted by attackers, for example, by identifying and exploiting vulnerabilities in IoT software and hardware, or their implementation, to facilitate unauthorized and malicious activities. Such devices have also been exploited to create a botnet network to generate Distributed Denial-of-Service (DDoS) traffic. DDoS represents a critical network-oriented cyberthreat, whose trend has been steadily rising over the last decade [1], [2]. For example, the DDoS attacks targeting Amazon AWS in Q1 of 2020 reportedly had a peak volume of 2.3 Tbps [3].

IoT devices and systems are found not only in an organizational or government setting but also in our homes (e.g., smart homes). Smart homes are one of the fastest-growing IoT applications, and the deployed devices are extremely heterogeneous. Such devices are often shipped with minimal or nonexistent security mechanisms, and in an effort to make these devices user friendly, the security requirements are often reduced [4]. In addition, most of the devices in a smart home are inexpensive and do not have significant computational capabilities and, consequently, they can be easily compromised to facilitate a broad range of nefarious activities, including generating DDoS traffic [5]. In a typical smart home ecosystem, there are several stakeholder groups, such as end users (homeowners or tenants within a home), Internet/telecommunication service providers, device manufacturers, and service providers (e.g., third-party service providers such as a monitored security service). These stakeholders generally have a vested interest not to be involved in malicious cyber activities, or for their devices, systems, platforms, and/or infrastructure to be exploited to facilitate nefarious activities. For example, it is in the interest of Internet/telecommunication service providers to promptly detect any unauthorized behavior/activities within a smart home environment, to protect their own network infrastructure and prevent the compromised devices/systems to be used as a launch pad against other devices and systems (with associated legal and financial implications).

A challenge is how to design an effective DDoS detection system that can be deployed in an increasingly diverse and dynamic smart home environment. For example, based on the generated network traffic characteristics, one might identify the types of devices commonly found in a smart home environment [5]. Along a similar line, a model for classifying IoT devices into previously defined classes was presented in our previous research [6], where we defined the classes of IoT devices-based purely on their traffic behavior and behavioral predictability (i.e., the coefficient of variation of the ratio of received and sent data). Building on this prior research, we present the following two hypotheses. First, it is possible to define profiles of legitimate (normal) traffic for classes of IoT devices, based on traffic flow characteristics. The second hypothesis is that, based on the individual class of IoT devices' legitimate traffic profiles, we can develop a supervised machine learning model that can effectively detect DDoS

Manuscript received January 25, 2021; revised March 31, 2021 and May 14, 2021; accepted June 15, 2021. Date of publication June 21, 2021; date of current version January 24, 2022. This work was supported by the University of Zagreb, Croatia, through the Project "Challenges of Information and Communication Networks, Technologies, Services and User Equipment in Establishing the Society 5.0 Environment—Phase 2" under Grant 210219; ZUID2020/2021. The work of Kim-Kwang Raymond Choo was supported by the Cloud Technology Endowed Professorship. *(Corresponding authors: Ivan Cvitić; Brij B. Gupta.)*

Ivan Cvitić and Dragan Peraković are with the Faculty of Transport and Traffic Sciences, Department for Information and Communication Traffic, University of Zagreb, 10000 Zagreb, Croatia (e-mail: ivan.cvitic@fpz.unizg.hr; dragan.perakovic@fpz.unizg.hr).

Brij B. Gupta is with the National Institute of Technology Kurukshetra, Kurukshetra 136119, India, and also with Asia University, Taichung 413, Taiwan (e-mail: bbgupta@nitkkr.ac.in).

Kim-Kwang Raymond Choo is with the Department of Information Systems and Cyber Security, University of Texas at San Antonio, San Antonio, TX 78249 USA (e-mail: raymond.choo@fulbrightmail.org).

Digital Object Identifier 10.1109/JIOT.2021.3090909

traffic as network anomalies generated from individual IoT devices. Hence, we develop a DDoS detection model for dynamic and heterogeneous IoT systems, which can be implemented in a smart home environment. We also remark that the DDoS detection model presented in this article uses a boosting method of logistic model trees (LMT), where a different version of the model is applied for each device class.

The contributions of our research can be summarized as follows.

1) The data set of legitimate IoT and anomalous DDoS traffic generated in this research will be made publicly available to the broad scientific community (and there is a lack of such data sets in [7]—see also the second section).
2) Our defined process of forming normal traffic profiles for classes of IoT devices.
3) Our proposed DDoS detection model, which uses device classes to detect DDoS traffic. We posit that such an approach is more effective, as we will also demonstrate later in this article.

The remainder of this article is organized as follows. Section II briefly reviews the related DDoS literature. Section III describes our data collection methodology, data set preprocessing, and DDoS detection model development based on the logistic model tree method from a supervised machine learning pool. Section IV shows the analysis of the findings, which show that the model accuracy is high for all classes of devices (i.e., accuracy rate between 99.92% and 99.99%). We will also discuss the implications of our work. In Section V, we will conclude this article and discuss future research possibilities.

II. Related Research

There have been many applications of machine learning techniques to detect DDoS traffic, which can be categorized into those based on supervised techniques (using existing knowledge to classify future unknown instances) and those based on unsupervised techniques (trying to determine the corresponding instance class without prior knowledge). For example, Doshi *et al.* [8] developed a model of binary classification of traffic on legitimate and DDoS traffic using five different machine learning methods. Specific features of Smart Home IoT (SHIoT) traffic were observed through changes in traffic characteristics, such as packet size, packet interim times, protocols used, and changes in the number of destination Internet protocol (IP) addresses with which these devices communicate at different time intervals. The research presented in [9] also proposed detecting DDoS traffic generated by IoT devices in a corporate environment, using the Deep Autoencoders method based on artificial neural networks. Özçelik *et al.* [10] suggested that DDoS traffic detection's efficiency is higher if it is performed at the edge of the observed IoT environment. Cvitić *et al.* [11] proposed a conceptual DDoS detection model that takes classes of IoT devices in consideration.

Despite the high accuracy of detection and the advantages in many of these existing approaches, several shortcomings and challenges remain. A key challenge is the lack of relevant data sets that can be used to train machine learning-based detection models [12], [13]. While there are a number of data sets containing DDoS and normal traffic, these are often obsolete and consequently reduce detection accuracy, because they do not reflect current traffic characteristics as newer devices, network concepts (e.g., software-defined networks), and services are been deployed [14], [15]. For example, Doshi *et al.* [8] used three devices with traffic collected over 10-m period, while the research in [9] used nine devices, of which five are webcams or security cameras. However, the data set from [9] is not publicly available in its original form. It is only available as a *.csv* file containing already extracted traffic features. This is limiting for other researchers because it does not possess the generated traffic in its original form stored in a format that would allow researchers to extract and calculate features that differ from those extracted by Meidan *et al.* [9]. Saharkhizan *et al.* [16] used data sets obtained by simulation in their proposed approach based on the long short-term-memory (LSTM) method to detect attacks in the IoT network. In [17], a two-level model was used to analyze network traffic flows. The traffic flow feature was selected empirically, and existing public data sets were used to evaluate the detection model. Salman *et al.* [18] presented a model for identifying IoT devices and the detection of attacks on IoT devices using several machine learning methods (i.e., decision tree, random forest, and deep learning methods). The study used a data set of traffic collected using seven IoT devices. The maximum detection accuracy of the developed model is 94.47%. Other DDoS detection approaches include those presented in [19]–[21]. Another observation from these works is that the data sets are generally very small and nonrepresentative of a real-world system.

Creating a robust testbed to generate realistic data sets is challenging, time consuming, and expensive partly due to the different possible configuration combinations. Existing data sets also differ in the way they are generated, which can be synthetic, simulated, or real [22]. Synthetic data sets are generated to meet the specific requirements and conditions that real data sets also meet. Existing data sets used in the literature are also generally dated (e.g., created between 1998 and 2012) and, hence, may not be representative of today's communication networks. Even newer data sets rarely have any IoT traffic included—see also Table I. Examples of the existing data sets include the one from the University of New South Wales in Australia [23], which comprises a number of SHIoT devices. For the development of an anomaly detection system, it is essential to have data sets containing normal/legitimate traffic generated by IoT devices. From such a data set, it is possible to define normal traffic behavior profiles for an individual device or a whole class of IoT devices.

III. Proposed Approach

A. Testbed Setup

The setup of our smart home laboratory environment is shown in Fig. 1, and also presented in [24]. Our environment

TABLE I
SNAPSHOTS OF EXISTING NON-IOT AND IOT DATA SETS

Dataset	Devices (number, types, etc.)	Setup (synthetic, simulated/emulated, or real)	Collected traffic	Year created	Types of traffic acquired
DARPA'98 [41]	Non-IoT, conventional devices (PC, servers) No. of devices: N/A	Simulated (small network) – simulated Air Force Base network	Collecting time: 2 weeks	1998	Attack traffic (38 types of attacks) DoS (11 types), R2L (unauthorized access from remote machine, 14 types) U2R (unauthorized access to local root, 7 types), probe (6 types)
KDDcup99 [42]	Non-IoT, convencional devices(PC, servers) No. of devices: N/A	Simulated traffic in a military environment (small network)	Attack instances - 3,925,650 Benign instances - 972,781	1999	DoS (SYN flood), R2L, U2R, probe
CAIDA [43]	Non-IoT, conventional devices (PC, servers) No. of devices: N/A	Real	Collecting time: 1 hour	2007	DDoS traffic
NSL-KDD [44]	Non-IoT, conventional devices (PC, servers) No. of devices: N/A	Emulated (small network)	Number of instances (train set): 4,898,431 Number of instances (test set): 311,027	2009	Attack traffic, normal/legitimate traffic
TUIDS [45]	Non-IoT, conventional devices (PC, servers) No. of devices: N/A	Emulated	Collecting time: 4 week (5.3 GB) No. of packets – 432875 No. of flows - 400131	2011/ 2012	Attack traffic (16 attack types) Normal traffic
CICIDS2017 [46]	Non-IoT, conventional devices (PC, servers) 25 users behviour profiles	Real	Collecting time: 24 hours (4.6 GB)	2017	Attack traffic (High-volume and low-volume application-level DDoS)
CSE-CIC-IDS2018 [47]	Non-IoT, conventional devices (PC, servers) No. of devices: 50 machines	Real	N/A	2018	Attack traffic (seven scenarios: Brute-force, Heartbleed, Botnet, DoS, DDoS, Web attacks, and infiltration of the network from inside)
N-BaIoT	IoT devices infected with Mirai and Bashlite No. of devices: 9	Real	Number of instances: 7062606	2018	Attack traffic (spam, UDP flood, TCP flood, Scan, ACK flood) Normal/legitimate traffic
Bot-IoT [48]	IoT devices No. of devices: 5 simulated IoT devices	Simulated (Ostinato and Node-red tool)	Collecting time: 4 week No. of packets – 432875 No. of flows - 400131	2019	Normal IoT, Attack IoT (Information gathering, DoS, Keylogger)
CICDDoS2019 [49]	Non-IoT, conventional devices (PC, servers)	Simulated	No. of instances for attack – 73360900 No. of instances for benign - 9543	2019	Attack traffic (PortMap, NetBIOS, LDAP, MSSQL, UDP, UDP-Lag, SYN, NTP, DNS, SNMP, SSDP, WebDDoS, TFTP
University of New South Wales [31]	IoT devices No. of devices: 28 real IoT devices	Real	Collecting time: 26 weeks Daily average traffic – 365 MB	2019	Normal/legitimate traffic

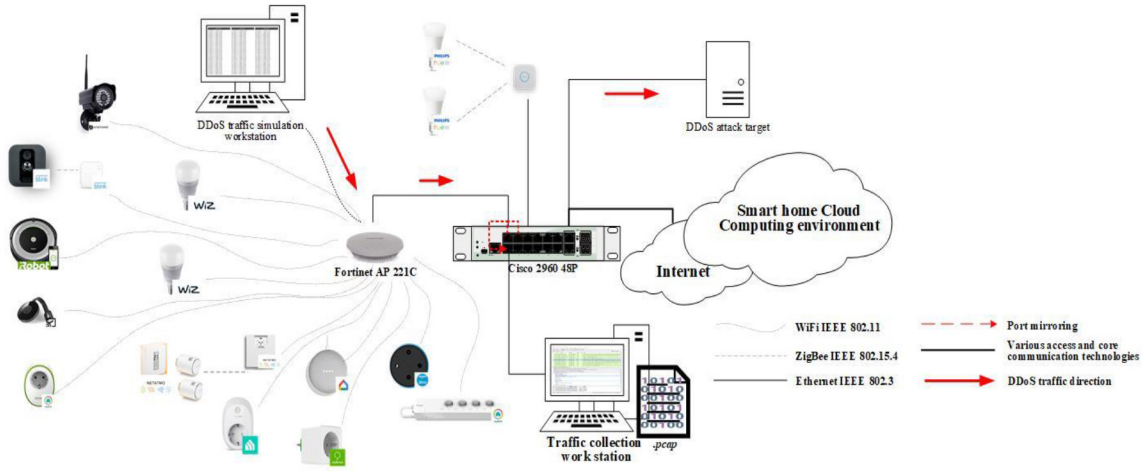


Fig. 1. Smart home testbed.

consists of 41 diverse SHIoT devices, and the underpinning communication infrastructure and software-hardware platform are also setup to enable traffic collection that can be used to train DDoS detection models.

In addition to the primary data collected in this research, we also used secondary data from [23], including a larger number of various SHIoT devices (i.e., greater device heterogeneity). The Fortinet AP 221C wireless access point, the Cisco 2960 Catalyst 48 Power over Ethernet (PoE) switch, the HP Pavillion dm1, and Microsoft HP 10 10.0.17134 build 17 134 workstations have been set up to capture traffic using port mirroring, x64 processor architecture, AMD E-350, 1600-MHz two cores, 4-GB RAM) with Wireshark software tool version 2.6.3 installed. The switch's physical communication ports (FA0/1 and FA0/3) to which the wireless access point and IoT hub for the Phillips Hue device are connected are configured for port mirroring. These ports are set up as sources, which ensures that all traffic to and from them are mirrored (mapped) to the destination contact port (FA0/2). A traffic collection workstation is connected to this port. With

TABLE II
ORIGINAL LEGITIMATE AND DDOS TRAFFIC DATA SETS' CHARACTERISTICS

	Number of files	Number of collected packets	The amount of data collected (GB)	Collection period (hours)
Primary (sum)	103	456,174,601	344.59	2,472.01
Secondary (sum)	41	99,334,088	38.16	986.45
DDoS-UDP	245	269,806,374	19.95	10.75
DDoS-TCP	73	85,373,401	5.88	17.12
DDoS-ICMP	195	217,593,439	16.1	8.75

a legitimate traffic profile of a SHIoT device, it is crucial to have a data set that includes DDoS traffic. These two sets form the basis for developing an effective model for detecting network traffic anomalies such as DDoS traffic generated by SHIoT devices.

Given that legitimate traffic comes from the primary and secondary sources, where the author does not have access to the secondary source devices, a key challenge is the manipulation of SHIoT devices to generate DDoS traffic. Therefore, in this research, for generating DDoS traffic BoNeSi (the open-source software tool) was used [25]. The virtual workstation was used to generate DDoS traffic and create a data set of illegitimate traffic. The virtual workstation's configuration is as follows: Linux Ubuntu 19.04 operating system with dedicated 4 GB of RAM, an Intel Core i7-5500U processor (4x2.40 GHz). In Fig. 1, the virtual machine and BoNeSi tool denote the SHIoT device in the local smart home network generating DDoS traffic. For practical reasons, the BoNeSi tool was used to simulate illegitimate traffic generated by the SHIoT device in order to minimize the risk of compromising the real device. BoNeSi is not just a network traffic generator (as the tool's documentation suggests), it is also a powerful and efficient DoS and DDoS generator and simulator tool. Hence, our choice for using it to simulate traffic similar to those generated by an individual SHIoT device as part of a botnet. In addition, the illegitimate traffic was generated in an isolated environment to avoid breaking the laws of the Republic of Croatia, the European Union, and the United States. For this research, the attack destination is less important than the attack source. Three types of DDoS traffic at the infrastructure level were generated and collected (UDP, TCP, and ICMP) as they are more frequent then attacks on the application layer.

In terms of the number of collected files containing 24-h cycles of generated traffic, the number of collected packets, the sum of collected data, and the overall time of data collection, the characteristics of the initially collected legitimate and DDoS traffic data are shown in Table II.

B. Defining Legitimate Traffic Profiles for Classes of SHIoT Device

As discussed earlier, SHIoT is a dynamic and ubiquitous environment, where new consumer IoT devices with different functionalities are constantly introduced to the market.

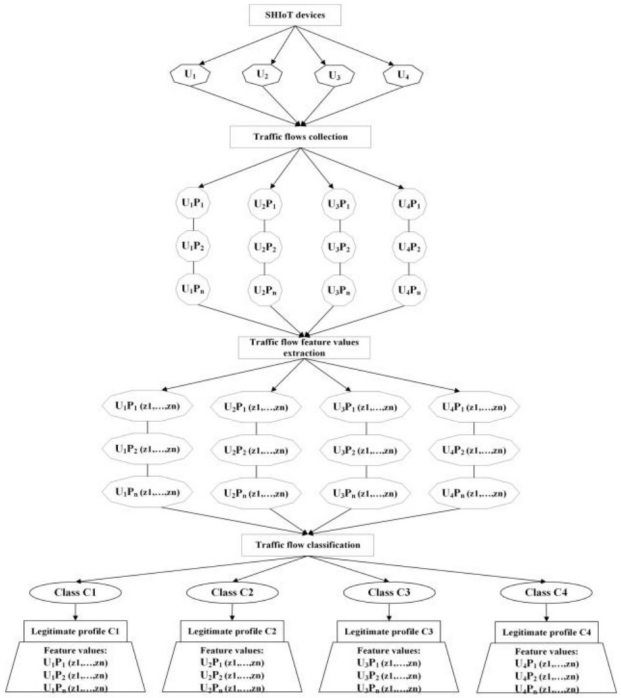


Fig. 2. Process of determining legitimate traffic profiles for SHIoT device classes.

Therefore, new, unknown SHIoT devices may have functionalities different from those of the currently available SHIoT devices.

This presents a challenge in identifying such devices and knowing their legitimate behavior, which forms the basis for detecting behavioral anomalies such as generating DDoS traffic.

In order to develop a DDoS traffic detection model based on the previously defined SHIoT device classes, it is necessary to define a legitimate traffic profile of each device class. In the development of any anomaly detection model based on supervised machine learning methods, it is necessary to have a set of data that will represent legitimate traffic and a set of data that will represent illegitimate traffic.

The defined classes of SHIoT devices [5] enable the establishment of a legitimate traffic profile of a particular class of devices, which is important in the later development of anomaly detection models. In doing so, the values of the traffic characteristics of the SHIoT device become part of the legitimate profile of the observed device class. The profile of legitimate traffic of a particular class of SHIoT devices is defined by the values of the characteristics of those traffic flows that are assigned by the classification model to a particular class of SHIoT devices, as shown in Fig. 2.

Let the SHIoT device be represented by U_x, and the traffic flow generated by such a device by U_xPT_y. Each device U_x is represented as a set of traffic flows U_xPT_y, i.e., each device contains a set of traffic flows, $U_x = \{U_xPT_1, \ldots, U_xPT_y\}$. Then, the legitimate traffic profile of each class C is defined as a set of traffic flows that are identified by the classification model as part of class C, ie $C_m = \{U_1PT_1, \ldots, U_xPT_y\}$; $m \in \{1, 2, 3, 4\}$. When each traffic flow is represented by

TABLE III
NETWORK TRAFFIC FLOW FEATURE DESCRIPTION

Feature name	ID	Feature description	Feature name	ID	Feature description
flowID	z1	Traffic flow ID	max_flowpktl	z42	Maximum length of a flow
srcIP	z2	Source IP address	mean_flowpktl	z43	Mean length of a flow
src_port	z3	Source communication port	std_flowpktl	z44	Standard deviation length of a flow
dstIP	z4	Destination IP address	min_flowiat	z45	Minimum inter-arrival time of packet
dst_port	z5	Destination communication port	max_flowiat	z46	Maximum inter-arrival time of packet
proto	z6	Used communication protocols in traffic flow	mean_flowiat	z47	Mean inter-arrival time of packet
timestamp	z7	Date ad time of traffic flow start	std_flowiat	z48	Standard deviation inter-arrival time of packet
Feduration	z8	Duration of the flow in Microsecond	flow_fin	z49	Number of packets with FIN
total_fpackets	z9	Total packets in the forward direction	flow_syn	z50	Number of packets with SYN
total_bpackets	z10	Total packets in the backward direction	flow_rst	z51	Number of packets with RST
total_fpktl	z11	Total size of packet in forward direction	flow_psh	z52	Number of packets with PUSH
total_bpktl	z12	Total size of packet in backward direction	flow_ack	z53	Number of packets with ACK
min_fpktl	z13	Minimum size of packet in forward direction	flow_urg	z54	Number of packets with URG
min_bpktl	z14	Minimum size of packet in backward direction	flow_cwr	z55	Number of packets with CWE
max_fpktl	z15	Maximum size of packet in forward direction	flow_ece	z56	Number of packets with ECE
max_bpktl	z16	Maximum size of packet in backward direction	downUpRatio	z57	Download and upload ratio
mean_fpktl	z17	Mean size of packet in forward direction	avgPacketSize	z58	Average size of packet
mean_bpktl	z18	Mean size of packet in backward direction	fAvgSegmentSize	z59	Average size observed in the forward direction
std_fpktl	z19	Standard deviation size of packet in forward direction	fAvgBytesPerBulk	z60	Average number of bytes bulk rate in the forward direction
std_bpktl	z20	Standard deviation size of packet in backward direction	fAvgPacketsPerBulk	z61	Average number of packets bulk rate in the forward direction
total_fiat	z21	Total time between two packets sent in the forward direction	fAvgBulkRate	z62	Average number of bulk rate in the forward direction
total_biat	z22	Total time between two packets sent in the backward direction	bAvgSegmentSize	z63	Average size observed in the backward direction
min_fiat	z23	Minimum time between two packets sent in the forward direction	bAvgBytesPerBulk	z64	Average number of bytes bulk rate in the backward direction
min_biat	z24	Minimum time between two packets sent in the backward direction	bAvgPacketsPerBulk	z65	Average number of packets bulk rate in the backward direction
max_fiat	z25	Maximum time between two packets sent in the forward direction	bAvgBulkRate	z66	Average number of bulk rate in the backward direction
max_biat	z26	Maximum time between two packets sent in the backward direction	sflow_fpacket	z67	The average number of packets in a sub flow in the forward direction
mean_fiat	z27	Mean time between two packets sent in the forward direction	sflow_fbytes	z68	The average number of bytes in a sub flow in the forward direction
mean_biat	z28	Mean time between two packets sent in the backward direction	sflow_bpacket	z69	The average number of packets in a sub flow in the backward direction
std_fiat	z29	Standard deviation time between two packets sent in the forward direction	sflow_bbytes	z70	The average number of bytes in a sub flow in the backward direction
std_biat	z30	Standard deviation time between two packets sent in the backward direction	min_active	z71	Minimum time a flow was active before becoming idle
fpsh_cnt	z31	Number of times the PSH flag was set in packets travelling in the forward direction (0 for UDP)	mean_active	z72	Mean time a flow was active before becoming idle
bpsh_cnt	z32	Number of times the PSH flag was set in packets travelling in the backward direction (0 for UDP)	max_active	z73	Maximum time a flow was active before becoming idle
furg_cnt	z33	Number of times the URG flag was set in packets travelling in the forward direction (0 for UDP)	std_active	z74	Standard deviation time a flow was active before becoming idle
burg_cnt	z34	Number of times the URG flag was set in packets travelling in the backward direction (0 for UDP)	min_idle	z75	Minimum time a flow was idle before becoming active
total_fhlen	z35	Total bytes used for headers in the forward direction	mean_idle	z76	Mean time a flow was idle before becoming active
total_bhlen	z36	Total bytes used for headers in the forward direction	max_idle	z77	Maximum time a flow was idle before becoming active
fPktsPerSecond	z37	Number of forward packets per second	std_idle	z78	Standard deviation time a flow was idle before becoming active
bPktsPerSecond	z38	Number of backward packets per second	Init_Win_bytes_forward	z79	The total number of bytes sent in initial window in the forward direction
flowPktsPerSecond	z39	Number of flow packets per second	Init_Win_bytes_backward	z80	The total number of bytes sent in initial window in the backward direction
flowBytesPerSecond	z40	Number of flow bytes per second	Act_data_pkt_forward	z81	Count of packets with at least 1 byte of TCP data payload in the forward direction
min_flowpktl	z41	Minimum length of a flow	min_seg_size_forward	z82	Minimum segment size observed in the forward direction

its characteristics z, it can be observed as a set of values of features that represent the observed traffic flow, $U_xPT_y = \{z(U_xPT_y)_1, \ldots, z(U_xPT_y)_n\}$.

In addition to the fact that it is possible to define classes of SHIoT devices (see [5]), it is also possible to classify devices, i.e., traffic flows generated by such devices using a developed classification model and traffic flow features with high classification accuracy (99.7956%). Such results enable creating a legitimate traffic profile of a particular class of SHIoT devices [6].

C. Formation of Data Sets for the Development of DDoS Traffic Detection Models

The SHIoT device classes defined by the research enable the identification of the class affiliation of the device based on the traffic flow generated by the device. This also enables the creation of a legitimate traffic profile because each traffic flow assigned to one of the four defined classes by the classification model becomes part of a set that represents a legitimate traffic profile of the same class. In order to develop a model for detecting (illegitimate) DDoS network traffic, the LMT method was used. For the implementation of the method and data processing, we used the WEKA software tool, as well as data sets that represent profiles of normal traffic resulting from the SHIoT device classification model and data sets of illegitimate DDoS traffic.

Four data sets (i.e., C1DDoS, C2DDoS, C3DDoS, and C4DDoS) containing the combined vectors of the legitimate traffic profile characteristics of each class of SHIoT devices and DDoS traffic were built. Initially, all four sets contain the values of all independent traffic flow features (83 in total) listed in Table III. For feature extraction, we used the CICFlowMeter tool [26]. The number and distribution of legitimate and DDoS traffic flows in the data sets were balanced and based on legitimate traffic profiles derived from the classification model of SHIoT devices shown in [5].

TABLE IV
PARTIAL PRESENTATION OF DATA SETS USED IN THE DEVELOPMENT OF A DDOS DETECTION MODEL

Feature vector	z8	z9	z10	z11	z12	…	z23	…	z36	…	z83	Class
						C1DDoS						
1	110,176,901	5	4	372	648	…	13,800,000	…	113,851	…	54,900,000	C1
2	114,974,487	13	1	0	0	…	8,844,190,000,000	…	0	…	5,402,915	DDoS
						C2DDoS						
1	32,421,069	12	12	2,973	6,626	…	1,409,612	…	76	…	0	C2
2	119,994,320	8	5	724	913	…	9,999,527	…	214,652	…	29,800,000	C2
						C3DDoS						
1	91,127,887	3	3	33	95	…	18,225,577	…	120,725	…	90,875,582	C3
2	47,780	5	5	84	474	…	5,309	…	2	…	0	C3
						C4DDoS						
1	119,436,915	16	18	5,158	527	…	3,619,300	…	136	…	6,167,447	C4
2	119,436,449	16	18	5,158	527	…	3,619,286	…	136	…	6,138,410	C4

As with any machine learning model development, the goal is to use those independent features, whose change has the greatest impact on changing the dependent feature. It is also important to reduce those features that can lead to model bias. Therefore, as with the development of the SHIoT device classification model, independent features z1–z7 represent traffic flow identifiers and contain information on the source and destination IP addresses, protocols used, and traffic flow generation time removed from the initial data sets. As a result, 76 independent features were obtained, which will be observed for further development of the model, and whose partial presentation is shown in Table IV. The table partially shows the data set used to develop the DDoS detection model. Each set consists of each traffic flow's values' independent features and the associated corresponding dependent feature that represents the class. In this case, the class is binary, i.e., it can take two values (0, 1), which indicates the traffic flow as legitimate for the observed class or illegitimate, i.e., the traffic flow created as a result of generating DDoS traffic.

This approach is necessary for further development of the model with the application of the method of supervised machine learning. We then leverage the LMT method in the development of our DDoS detection model. The LMT method, developed in 2003 [27], is a boosting method of supervised machine learning that is a fusion of two commonly used classification methods: 1) logistic regression and 2) decision trees, to upgrade them. The method's basic working principle consists of creating decision trees and forming a logistic regression model at the tree nodes. Logistic regression models build on each other into a single model. In this way, the logistic regression method estimates the probability of belonging of an individual feature vector to a defined class. For numerical features (such as those found in the presented data sets), the feature representing the node in which the division is the "purest" is selected. This implies that the maximum number of feature vectors belongs to one class when the selected feature's value is below the defined value threshold and to another class if the selected feature is observed above the defined value threshold. The LMT model consists of a decision tree structure containing internal nodes N and a set of terminal node T. S representing an entire data set with all features [28]. The decision tree then divides the set S into disjoint subsets (regions) of S_t. Each region is represented by a terminal node of the tree as shown by the following:

$$S = \bigcup_{t \in T} S_t, S_t \cap S_{t'} = \emptyset \text{ for } t \neq t' \tag{1}$$

where

S set of all feature vectors;
S_t disjoint subset of feature vectors;
t terminal node from a set of terminal nodes T.

Unlike the classical decision tree, the LMT method associates logistic regression functions, f_t instead of the class designation, with terminal nodes $t \in T$. The logistic regression considers the subset $Z_t \subseteq Z$ of all independent features in the data set and models the probability of belonging to the class according to

$$\Pr(G = j | X = x) = \frac{e^{F_j(x)}}{\int_{k=1}^{J} e^{F_k(x)}} \tag{2}$$

$$F_j(x) = \alpha_0^j + \sum_{k=1}^{m} \alpha_{z_k}^j \cdot z_k \tag{3}$$

where

α^j coefficient of independent feature z;
z_k independent feature from a set of independent features $Z = \{z_1, \ldots, z_m\}$.

The final LMT model takes the form given by

$$f(x) = \sum_{t \in T} f_t(x) \cdot I(x \in S_t), I(x \in S_t) \begin{cases} 1 & \text{if } x \in S_t \\ 0 & \text{else.} \end{cases} \tag{4}$$

According to Landwehr *et al.* [28], the goal of the method is to adapt the data so that the logistic decision tree is generalized (pruned) to the level of one logistic regression model, i.e., to the root node of the decision tree if possible, given the data set over by which the method is applied.

Selecting the relevant independent features from the set of all features when using the LMT method does not need to be

undertaken separately as this method adjusts (fitting) regression function to each independent feature using the least square error. According to this criterion, the final model includes those features that result in the smallest square error, as shown in Table V. Using the WEKA software environment, the described LMT method was implemented on all our four data sets (i.e., C1DDoS, C2DDoS, C3DDoS, and C4DDoS) to develop LMT models for each class of SHIoT devices.

1) LMT Model for C1 Class of SHIoT Devices: By implementing the LMT method using the WEKA programming environment, independent features with the greatest influence on the dependent feature were selected, and a logistic regression model was developed since the decision tree is generalized to the root node. Therefore, at the decision tree's root node, the corresponding LMT model is defined

$$\Pr(G = C1|X = x) = \frac{e^{F_{C1}(x)}}{e^{F_{C1}(x)} + e^{F_{\mathrm{DDoS}}(x)}} \tag{5}$$

$$\Pr(G = \mathrm{DDoS}|X = x) = \frac{e^{F_{\mathrm{DDoS}}(x)}}{e^{F_{C1}(x)} + e^{F_{\mathrm{DDoS}}(x)}}. \tag{6}$$

Both F_{C1} and F_{DDoS} functions were used to determine the probability of belonging to a class by modeling independent features' influence on the dependent feature. For class C1, the logistic regression model takes the form shown by

$$\begin{aligned} F_{C1}(x) = {} & -1.37 + 0.02 \cdot \mathrm{z14} + 0.01 \cdot \mathrm{z18} + 3.29 \cdot \mathrm{z38} \\ & + 0.01 \cdot \mathrm{z46} + (-3,72) \cdot \mathrm{z50} + (-1.08) \cdot \mathrm{z51} \\ & + (-0.2) \cdot \mathrm{z54} + 0.88 \cdot \mathrm{z58} + 0.57 \cdot \mathrm{z74} \end{aligned} \tag{7}$$

$$\begin{aligned} F_{\mathrm{DDoS}}(x) = {} & -F_{C1}(x) = 1.37 + (-0.02) \cdot \mathrm{z14} + (-0.01) \\ & \cdot \mathrm{z18} + (-3.29) \cdot \mathrm{z38} + (-0.01) \cdot \mathrm{z46} + 3.72 \\ & \cdot \mathrm{z50} + 1.08 \cdot \mathrm{z51} + 0.2 \cdot \mathrm{z54} + (-0.88) \cdot \mathrm{z58} \\ & + (-0.57) \cdot \mathrm{z74}. \end{aligned} \tag{8}$$

The model includes independent features for which the method of least square deviation determined the greatest influence on the change of the dependent feature. The effect of the independent on the dependent feature is defined by sufficient coefficients for each feature. The assigned coefficient indicates that one unit of the independent feature's increase will change the dependent feature by the logarithm of the logistic regression coefficients' layout, while the other independent variables will remain unchanged. For example, the coefficient assigned to the independent feature z14 is -0.02 and represents an estimate of the change (increase or decrease; as determined by the sign), in this case, a decrease in the amount of logarithm of the dependent feature if the independent feature z14 increases by one unit and the others the independent features in the model remain unchanged.

2) LMT Model for C2 Class of SHIoT Devices: The LMT model of DDoS detection for class C2 SHIoT devices was developed in the same way as the previously described model for class C1. Since different SHIoT devices belong to different classes, it is intuitively clear that the traffic flows generated by SHIoT devices of class C2 differ in terms of feature values from the traffic flows of SHIoT devices of class C1. Therefore, the model developed for this class of devices, although based on the same method, has certain differences. This primarily refers to the decision tree's appearance and the independent features included in the model, and the coefficients added to these features. This means that the independent features that affect the dependent feature change differ from class to SHIoT devices class.

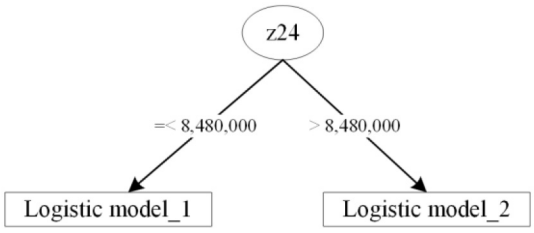


Fig. 3. Example of application of the LMT method in the classification of feature vectors.

On the other hand, different classes may share the same relevant independent features, but they also have different coefficients with different degrees of influence. For class C2, the decision tree differs from that for class C1 because it is impossible to define logistic regression models at the root node that would provide satisfactory LMT model performance. In this case, the decision tree is generalized to three nodes (one root node and two terminal nodes), as shown earlier in Fig. 3. Therefore, two logistics models are defined at the terminal nodes. LM1, according to expressions (9) and (10) and LM2, according to expressions (11) and (12), which are applied depending on the condition that is satisfied when branching the decision tree

$$\begin{aligned} F_{C2}(x) = {} & -16.07 + 3.42 \cdot \mathrm{z10} + 4.35 \cdot \mathrm{z38} + 0.01 \cdot \mathrm{z41} \\ & + 0.01 \cdot \mathrm{z46} + (-2.06) \cdot \mathrm{z50} + (-0.39) \cdot \mathrm{z51} \\ & + 2.28 \cdot \mathrm{z54} + 0.97 \cdot \mathrm{z58} + 14.58 \cdot \mathrm{z74} \end{aligned} \tag{9}$$

$$\begin{aligned} F_{\mathrm{DDoS}}(x) = {} & -F_{C2}(x) = 16.07 + (-3.42) \cdot \mathrm{z10} + (-4.35) \\ & \cdot \mathrm{z38} + (-0.01) \cdot \mathrm{z41} + (-0.01) \cdot \mathrm{z46} + 2.06 \\ & \cdot \mathrm{z50} + 0.39 \cdot \mathrm{z51} + (-2.28) \cdot \mathrm{z54} + (-0.97) \\ & \cdot \mathrm{z58} + (-14.58) \cdot \mathrm{z74} \end{aligned} \tag{10}$$

$$\begin{aligned} F_{C2}(x) = {} & -20.68 + 2.32 \cdot \mathrm{z38} + 0.01 \cdot \mathrm{z46} + (-2.06) \cdot \mathrm{z50} \\ & + (-0.39) \cdot \mathrm{z51} + 2.28 \cdot \mathrm{z54} + 0.84 \cdot \mathrm{z58} \end{aligned} \tag{11}$$

$$\begin{aligned} F_{\mathrm{DDoS}}(x) = {} & -F_{C2}(x) = 20.68 + (-2.32) \cdot \mathrm{z38} + (-0.01) \\ & \cdot \mathrm{z46} + 2.06 \cdot \mathrm{z50} + 0.39 \cdot \mathrm{z51} + (-2.28) \cdot \mathrm{z54} \\ & + (-0.84) \cdot \mathrm{z58}. \end{aligned} \tag{12}$$

It is noted that the LMT model for detection of network traffic anomalies for SHIoT devices belonging to class C2 consists of a decision tree on whose terminal nodes there are two logistics models, and their use depends on which condition satisfies the observed feature vector concerning the value of independent feature z24. It also depends on this condition in which independent features will be included in the logistics model and the coefficients associated with these features.

3) LMT Model for C3 Class of SHIoT Devices: For the class C3 SHIoT devices to detect network traffic anomalies, an LMT model was developed on principle applied to classes C1 and C2. As for class C1, the decision tree is generalized to the root node to which one logistic model is associated.

TABLE V
DISPLAY OF INDEPENDENT FEATURES INCLUDED IN THE LMT

LMT model									
LMT-C1	LMT-C2		LMT-C3	LMT-C4					
Logistics models									
LM1	LM1	LM2	LM1	LM1	LM2	LM3	LM4	LM5	LM6
z14	z10	z38	z14	z10	z10	z10	z10	z10	z10
z18	z38	z46	z38	z16	z16	z16	z16	z16	z38
z38	z41	z50	z45	z20	z20	z20	z20	z38	z50
z46	z46	z51	z46	z36	z36	z36	z38	z41	z51
z50	z50	z54	z50	z38	z38	z38	z41	z42	z54
z51	z51	z58	z51	z41	z41	z41	z42	z45	z58
z58	z54		z54	z42	z42	z42	z45	z50	z74
z74	z58		z58	z45	z43	z44	z50	z51	
	z74		z74	z50	z44	z45	z51	z54	
				z51	z45	z46	z54	z58	
				z54	z46	z50	z58	z74	
				z58	z50	z51	z74		
				z74	z51	z58			
					z54	z73			
					z58	z74			
					z74				

The final form of the LMT model, with the most significant independent features and coefficients for class C3, is shown by

$$\begin{aligned} F_{C3}(x) &= -1.01 + 0.03 \cdot \text{z14} + 2.91 \cdot \text{z38} + 0.01 \cdot \text{z45} \\ &\quad + 0.02 \cdot \text{z46} + (-2) \cdot \text{z50} + (-1.82) \cdot \text{z51} \\ &\quad + 1.12 \cdot \text{z54} + 0.87 \cdot \text{z58} + 0.04 \cdot \text{z74} \qquad (13) \\ F_{\text{DDoS}}(x) &= -F_{C3}(x) = 1.01 + (-0.03) \cdot \text{z14} + (-2.91) \cdot \text{z38} \\ &\quad + (-0.01) \cdot \text{z45} + (-0.02) \cdot \text{z46} + 2 \cdot \text{z50} \\ &\quad + 1.82 \cdot \text{z51} + (-1.12) \cdot \text{z54} + (-0.87) \cdot \text{z58} \\ &\quad + (-0.04) \cdot \text{z74}. \qquad (14) \end{aligned}$$

The model included a total of nine independent (z14, z38, z45, z46, z50, z51, z54, z58, z74) features that were determined by the method of least squares to have the greatest impact on the change of the dependent feature.

4) LMT Model for C4 Class of SHIoT Devices: Class C4 devices, due to the higher C_u index, generate traffic and traffic flows whose characteristics are more difficult to distinguish from network traffic anomalies such as DDoS traffic.

The lower level of traffic predictability is caused by the device's mode of operation, such as a high level of user interaction, playback of audio/video content, and the like. This results in a more complex LMT model that cannot be generalized to the root node, but it consists of 11 nodes or six terminal nodes. A logistic regression model is defined on each branch of the decision tree ending in the terminal node.

In the present case, this means that the LMT model consists of a total of five branching points and six logistic regression models. An LMT model containing a decision tree and associated logistic regression models with selected relevant independent features and associated coefficients, as shown in Fig. 4.

D. Working Principle of the Developed Model for Detection of Illegitimate DDoS Network Traffic

The work of the developed model of illegitimate DDoS traffic detection takes place in two phases. The first phase is a prerequisite for later detection of DDoS traffic in the second phase and involves the classification of SHIoT devices based on generated traffic flow. The multiclass classification model results show that the SHIoT device can be classified into one of the four predefined classes concerning the traffic flows it generates with an accuracy of 99.79%.

After the device is successfully classified, the newly generated traffic flows are checked based on the LMT model for detecting illegitimate DDoS traffic, which determines whether these traffic flows belong to a recognized class or represent an anomaly of network traffic.

The basis for the development of the DDoS traffic detection model for a particular class is the profile of legitimate traffic of a particular class, resulting from the work of a multiclass classification model in the first phase.

In doing so, the values of traffic flow classified into certain predefined classes also become part of the profile of legitimate traffic of these classes. Depending on the corresponding class of SHIoT devices, an individual LMT model can detect deviations or anomalies from the existing normal traffic profile with high accuracy (LMT-C1 = 99.99%, LMT-C2 = 99.92%, LMT-C3 = 99, 97%, and LMT-C4 = 99.95%) and using different sets of independent traffic flow characteristics.

IV. RESULTS ANALYSIS AND DISCUSSION

The development of a DDoS detection model based on traffic characteristics and device class indicates the importance of recognizing the class to which the SHIoT device belongs as a fundamental activity of further recognizing anomalies in network traffic such as DDoS traffic. According to the model presented in the previous section, it is clear that not all independent features are equally important in detecting anomalies for a particular class. Likewise, certain features in one class may be relevant while viewed from the aspect of another class they do not have to.

An example is seer each class differs according to the number of relevant independent features, and it is also evident that the same features are not relevant in the detection of anomalies for each class.

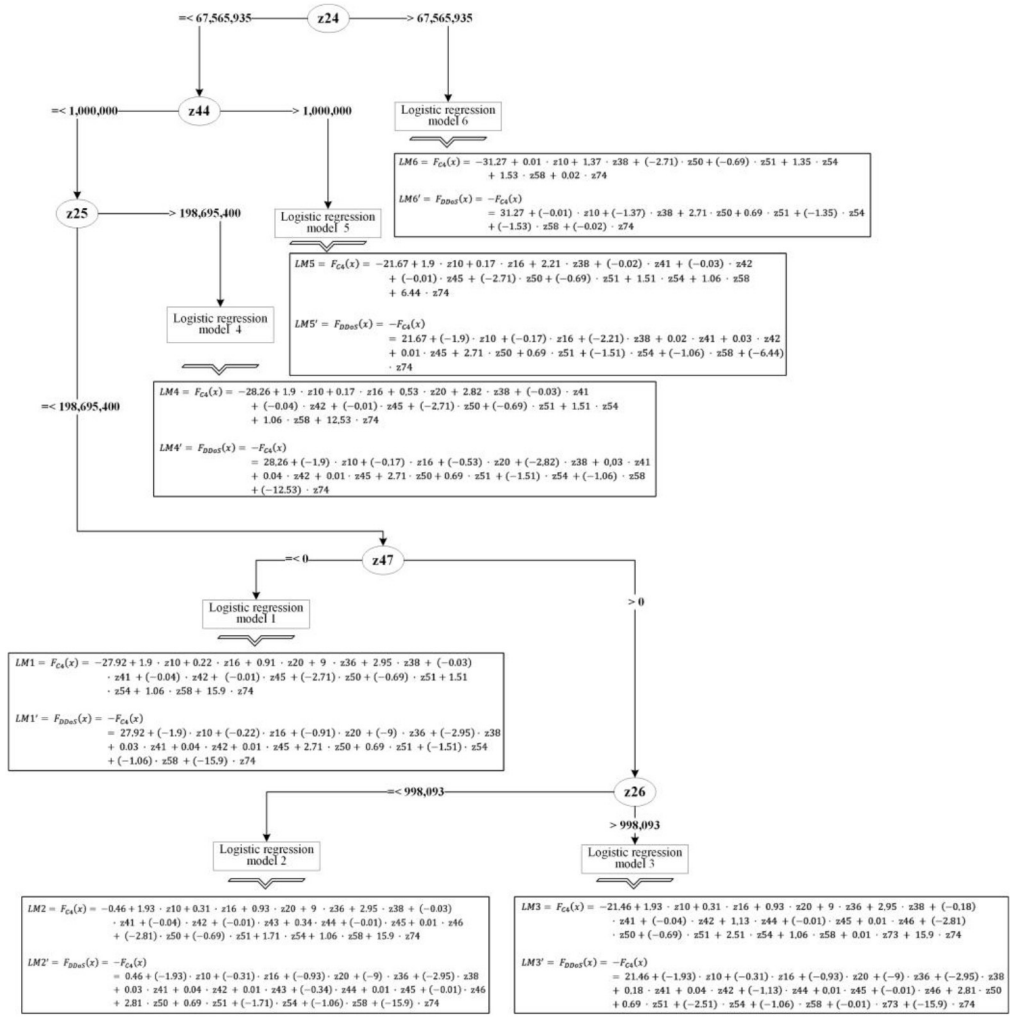


Fig. 4. LMT model of the DDoS detection model for class C4.

Furthermore, an individual independent feature's threshold value that determines the decision tree's branching differs for individual classes. As shown in Figs. 3 and 4, branching in the decision tree occurs based on the threshold value of the feature z24, representing the standard deviation of the

The interarrival packet times in the observed traffic flow expressed in microseconds ($\hat{A}\mu s$). In doing so, the algorithm C4.5 is used, which selects the threshold value of the independent feature that allows the purest division of the feature vector in the set [29]. Thus, for example, the threshold value of the z24 feature in the LMT model for class C2 differs from the threshold value of the same feature for class C4.

To evaluate the behavior of the model over data not included in the learning process, each version of the LMT model was validated using the k-fold cross-validation approach with $k = 10$. Cross-validation is a mathematical technique for evaluating the success of machine learning models on new, unknown data. This approach is used to test the model's output on data that was not used during the learning process. The model is iteratively extended k times over the data set in this way. The data set is split into k sections in each iteration. The remaining $k - 1$ portions of the set are grouped into a subset for model learning, while one part of the set is used to test the model [30]. Validation metrics (accuracy, kappa statistics, true-positive rate (TPR), false-positive rate (FPR), precision, F-measure, ROC-Receiver Operating Characteristics, and PRC-Precision-Recall Curve) are often used to test machine learning classification models.

TABLE VI
ACCURACY OF DEVELOPED MODELS AND KAPPA COEFFICIENT

Model	**LMT-C1**		**LMT-C2**	
Accurately classified examples	56,092	99.921%	59,660	99.996%
Misclassified examples	44	0.0784%	2	0.0034%
***Kappa* coefficient (κ)**	0.9984		0.9999	
Total	56,136		59,662	
Model	**LMT-C3**		**LMT-C4**	
Accurately classified examples	58,646	99.974%	59,879	99.958%
Misclassified examples	15	0.0256%	25	0.0417%
***Kappa* coefficient (κ)**	0.9995		0.9992	
Total	58,661		59,904	

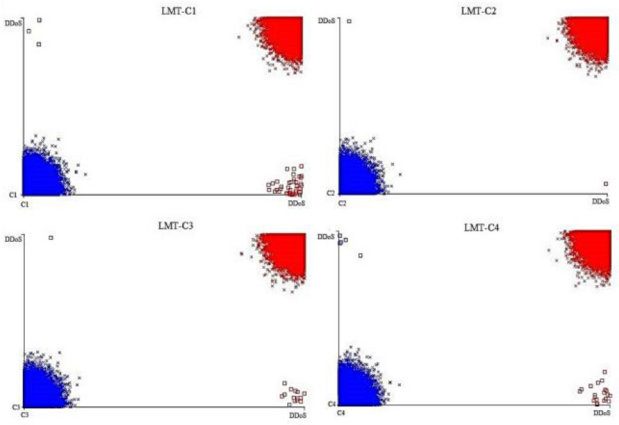


Fig. 5. Error visualization of LMT classification models for the corresponding classes.

A. Accuracy of Developed LMT Classification Models

True-positive (TP) examples, true-negative (TN) examples, false-positive (FP) examples, and false-negative (FN) examples reflect the share of correctly classified examples in the set of all examples

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{15}$$

where

- Acc proportion of accurately classified examples in the set of all examples;
- TP number of true positive examples;
- TN number of true negative examples;
- FP number of false positive examples;
- FN number of false negative examples.

According to the classification's accuracy, all four models show high performance, which means that based on the observed flow, they can determine with high accuracy whether the traffic flow is the result of legitimate communication of the device, or the device generates DDoS traffic. According to Table VI, the high accuracy of all four versions of the LMT model developed for each class of SHIoT devices can be observed. Errors in the classification of all four versions of the LMT model were visualized and shown in Fig. 5.

Fig. 5 shows that the detection model is most accurate for class C2 and the lowest performance is observed in the LMT-C1 model. From the given figure, it is observed that errors for all four models are prevalent for classifying DDoS traffic instances, indicating the need for better modeling of this class in future research.

To more clearly show the accuracy of the classification, a confusion matrix was used for all developed versions of the model. The confusion matrix is a performance metric for machine learning classification models with two or more classes as output, and it serves as the foundation for other metrics. Thus, the LMT model for device class C1 shows an accuracy of 99.9216%, or 56 092 accurately classified traffic flows, as a DDoS or traffic flow that legitimately belongs to a SHIoT device from class C1. A total of 44 traffic flows were misclassified, i.e., 0.0784% in the total set of 56 136. Out of 44 incorrectly classified traffic flows, 41 were predicted to belong to the legitimate traffic flow of class C1, while three traffic flows were classified as DDoS traffic, as shown by the confusion matrix in Table VII.

TABLE VII
CONFUSION MATRIX OF LMT MODELS FOR CLASSES C1 AND C2

Predicted class affiliation			
Class C1	**DDoS**		
28,065	3	**Class C1**	*Actual class affiliation*
41	28,027	**DDoS**	
Predicted class affiliation			
Class C2	**DDoS**		
29,830	1	**Class C2**	
1	29,830	**DDoS**	

TABLE VIII
CONFUSION MATRIX OF LMT MODELS FOR CLASSES C3 AND C4

Predicted class affiliation			
Class C3	**DDoS**		
29,329	1	**Class C3**	*Actual class affiliation*
14	29,317	**DDoS**	
Predicted class affiliation			
Class C4	**DDoS**		
29,947	5	**Class C4**	
20	29,932	**DDoS**	

In addition to high accuracy, the LMT model for device class C1 also shows a kappa coefficient ($\kappa = 0.9984$), which indicates high model performance.

The LMT model version developed for class C2 shows high accuracy (99.9966%), shown in Table VI. This implies 59 660 accurately classified traffic flows in a set of 59 662 traffic flows. The classification error is 0.0034%, i.e., two traffic flows, with one incorrectly assigned to class C2 and the other to DDoS traffic, which is evident from the confusion matrix shown in Table VII. The amount of kappa coefficient is 0.9999, which indicates a high success of this version of the LMT model.

The LMT classification model developed for class C3 provides an accuracy of 99.9744%, as shown in Table VI. Therefore, out of 58 661 traffic flows, 15 are misclassified, or 0.0256%, while 58 646 are accurately classified. According to the confusion matrix shown in Table VIII, one traffic flow was misclassified as DDoS traffic, while 14 traffic flows were misclassified as part of a legitimate class C3 traffic.

The amount of kappa coefficient of 0.9995, as with previous versions of the LMT model, indicates its high performance. The latest version of the LMT model, developed for class C4, shows an accuracy of 99.9583% which implies 59 879 correctly classified traffic flows. Therefore, 25 traffic flows were misclassified, five as DDoS traffic and 20 as legitimate class C4 traffic, as shown by the confusion matrix in Table VIII. The success of the model measured by the kappa coefficient is 0.9992, seen in Table VI.

B. Performance Analysis Based on Positive and Negative Model Results

Further analysis and performance evaluation of the developed LMT models was conducted using metrics based

TABLE IX
OVERVIEW OF LMT MODEL VALIDATION MEASURES (TPR AND FPR)

Class	True positive rate (TPR)			
	LMT-C1	LMT-C2	LMT-C3	LMT-C4
C1/C2/C3/C4	1	1	1	1
DDoS	0.999	1	1	0.999
Class	**False positive rate (FPR)**			
	LMT-C1	LMT-C2	LMT-C3	LMT-C4
C1/C2/C3/C4	0.001	0	0	0.001
DDoS	0	0	0	0

TABLE X
OVERVIEW OF LMT MODEL VALIDATION MEASURES (PRECISION AND F-MEASURE)

Class	Precision			
	LMT-C1	LMT-C2	LMT-C3	LMT-C4
C1/C2/C3/C4	0.999	1	1	0.999
DDoS	1	1	1	1
Class	**F-measure (F1 score)**			
	LMT-C1	LMT-C2	LMT-C3	LMT-C4
C1/C2/C3/C4	0.999	1	1	1
DDoS	0.999	1	1	1

on positive and negative results. Given that each measure has advantages and disadvantages, the success of a classification model based on machine learning must be conveyed across many successive methods.

The first such measure is the rate of TPR. The TPR reflects correctly categorized examples of a class in the set of all examples attributed to that class

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{16}$$

In the above equation, TPR is the true positive rate.

Table IX shows the TPR results for all versions of the LMT model with TPR for all legitimate traffic classes being 1. The TPR values for the DDoS class in LMT-C2 and LMT-C3 models are 1. LMT-C1 and LMT-C4 models notice a minimal drop in performance with a TPR of 0.999. The next important performance evaluation measure is the FP example rate (FPR) shown in the same table.

The FP rate represents the ratio of misclassified class examples in the set of all examples assigned to that class to (17). According to this measure, all models show good results for the legitimate traffic classes and the DDoS class

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{17}$$

In the above equation, FPR is the false positive rate.

According to (18), the precision calculation is used to express the number of correctly categorized examples in relation to the total number of examples belonging to that class.

According to (19), the F-measure or *F1* score represents the harmonic mean of the precision and the TPR [30]. The harmonic mean is more intuitive than the classical arithmetic mean for calculating the ratio's mean, according to [31]

$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{18}$$

TABLE XI
OVERVIEW OF LMT MODEL VALIDATION MEASURES (ROC AND PRC)

Class	ROC			
	LMT-C1	LMT-C2	LMT-C3	LMT-C4
C1/C2/C3/C4	0.999	1	1	1
DDoS	0.999	1	1	1
Class	**PRC**			
	LMT-C1	LMT-C2	LMT-C3	LMT-C4
C1/C2/C3/C4	0.998	1	1	1
DDoS	0.999	1	1	1

In the above equation, PPV is the positive prediction value

$$F1 = \frac{2(\text{PPV} \cdot \text{TPR})}{\text{PPV} + \text{TPR}}. \tag{19}$$

According to the values shown in Table X, both measures indicate high performance of all versions of the LMT model. A minimal drop in performance is observed for LMT-C1 and LMT-C4 (0.999) for classes C1 and C4 for the precision measure and for LMT-C1 for class C1 and DDoS for F1 rating (0.999).

All four versions of the LMT model's high performance are visible from the implemented ROC and PRC measures whose results are visible in Table XI. As one of the most important and most frequently used measures showing the classification model's quality, the ROC measure results indicate high quality of all versions of the developed LMT model. Proof of this is the TPR and TNR rate ratio value, which is 1 for the models LMT-C2, LTM-C3, and LMT-C4, and 0.999 for the model LMT-C1.

Since the data sets are stratified, the PRC measure, as an alternative to the ROC measure, which can better assess the impact of a large number of negative examples on model performance, gives almost equal values for all observed LMT models.

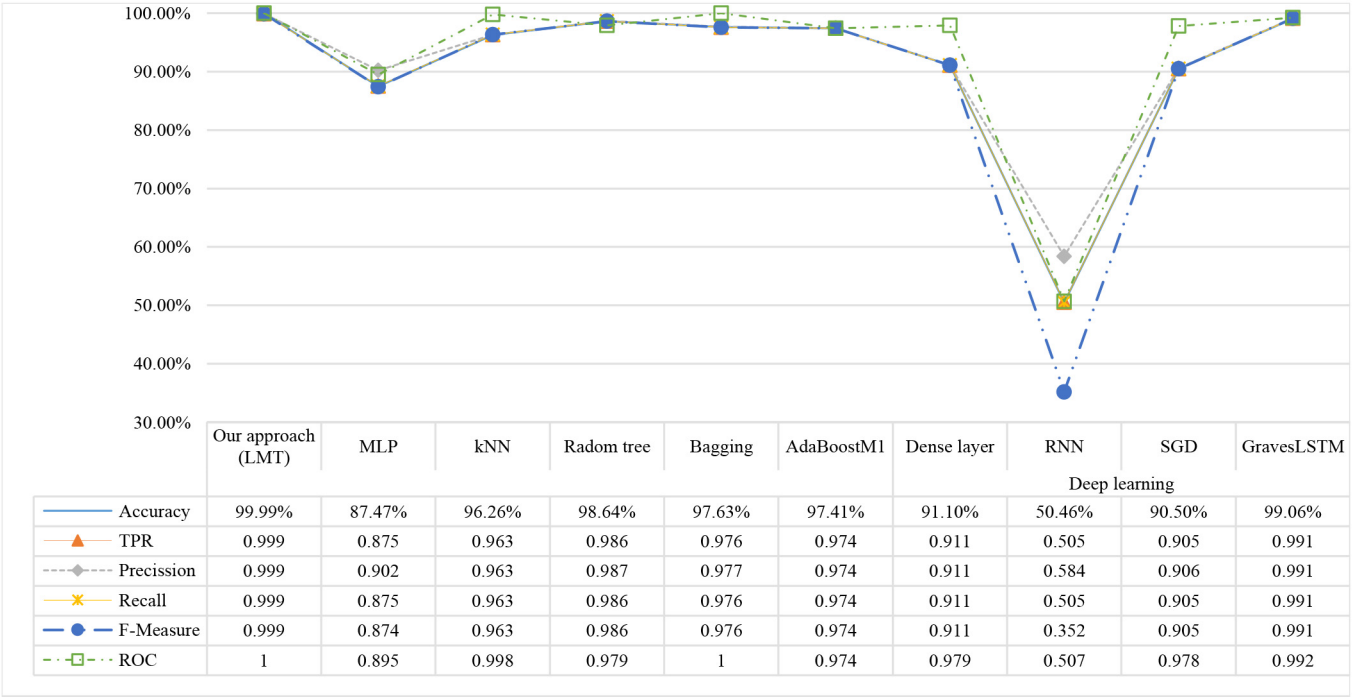
The presented results of the developed model confirm the second hypothesis of this research. Based on the defined profile of legitimate traffic of a particular class of IoT devices in a smart home environment, detecting illegitimate traffic generated by such devices is possible.

A comparative summary is shown in Table XII, and one can observe that our approach achieves the highest accuracy, precision, recall, and F-measure. It is also observed that we consider the highest number of SHIoT devices, and a key benefit of our approach is its ability to detect anomalous traffic generated by previously unseen IoT devices.

In summary, to evaluate the effectiveness of the LMT method applied in this study, we applied several frequently used machine learning methods over the same data set. Specifically, we compared the performance of our proposal with those of multilayer perceptron (MLP), k-Nearest Neighbors (kNN), Random Tree (RT), Bagging, AdaBoostM1, stochastic gradient descent (SGD), dense layer, Recurrent Neural Network (RNN), and GravesLSTM, in terms of accuracy, TPR, Precision, Recall, F-measure, and ROC. For implementing mentioned methods, we used WekaDeeplearning4j package for WEKA platform [32]. From the comparison results presented in Fig. 6, one can see that our approach generally outperforms the other applied methods.

TABLE XII
COMPARISON WITH OTHER COMPETING APPROACHES

Research	Used method	Dataset	Acuracy	Precision	Recall	F1 - Measure	Reqires profiling each individual device to detect anomaly
[16]	LSTM	Modbus network traffic data	0.9962	0.9935	0.9941	0.993	n/a
[18]	RNN, ResNet, ConvNet	7 real IoT devices	0.852 - 0.9997	0.6248 - 0.9997	0.6740 - 0.9997	0.6208 - 0.9997	no
[19]	J48	8 real IoT devices	n/a	0.9	0.899	0.888	yes
[42]	Convolutional neural networks (CNN)	CICDDoS2019	n/a	0.87	0.86	0.86	n/a
[43]	shingling-based graph sketching	28 real IoT devices	n/a	0.98	0.92	0.92	yes
[8]	KN, LSVM, DT, RF, NN	3 real IoT devices	0.991-0.999	0.983 - 0.999	0.870 - 0.999	0.927-0.999	yes
[44]	LGBM, DNN, SVM	8 real IoT devices / 5 Non IoT devices	n/a	n/a	0.370 - 1	n/a	yes
Our approach	Bosting based LMT	**41** real IoT devices (555,508,689 packets, 393.33 GB of traffic, 3,458.47 hours)	**0.9921 - 0,9996**	**0.999 - 1**	**0.999 - 1**	**0.999 - 1**	**no** (works with previously unseen devices)



	Our approach (LMT)	MLP	kNN	Radom tree	Bagging	AdaBoostM1	Dense layer	RNN	SGD	GravesLSTM
							Deep learning			
Accuracy	99.99%	87.47%	96.26%	98.64%	97.63%	97.41%	91.10%	50.46%	90.50%	99.06%
TPR	0.999	0.875	0.963	0.986	0.976	0.974	0.911	0.505	0.905	0.991
Precission	0.999	0.902	0.963	0.987	0.977	0.974	0.911	0.584	0.906	0.991
Recall	0.999	0.875	0.963	0.986	0.976	0.974	0.911	0.505	0.905	0.991
F-Measure	0.999	0.874	0.963	0.986	0.976	0.974	0.911	0.352	0.905	0.991
ROC	1	0.895	0.998	0.979	1	0.974	0.979	0.507	0.978	0.992

Fig. 6. Comparison of our approach with other competing machine learning and deep learning methods.

V. CONCLUSION AND FUTURE WORK

The DDoS detection model presented in this article deviates from the typical network traffic anomaly detection approaches. For example, prior approaches are largely based on generating a legitimate traffic profile that is assumed to apply to all terminal devices. Such an approach is logical in environments comprising conventional devices, whose traffic generates characteristics that are reflective of the operation of the installed applications on the devices and the way the users use such devices.

However, inexpensive IoT devices are somewhat limited in terms of their functionality, which is reflected in the characteristics of the traffic they generate. There are also IoT devices which are more computationally capable. Hence, existing non-IoT approaches may not be suitable, partly due to the diversity of IoT devices (and consequently, behavior). In other words, some devices will always generate similar traffic, while other devices that are capable of supporting greater interactions with the user may generate traffic that is irregular. Compounding this challenge is the significant growth in the number of devices in an IoT environment.

In other words, DDoS detection approaches based on individual device characteristics require relearning or even redevelopment of the underlying model for each new device that appears on the market. Such an approach is extremely complex and insufficiently generic in an increasingly complex and

dynamic IoT environment. This is limitation we attempted to address in this article.

Our approach assumes that there is no one overarching legitimate traffic profile for IoT devices, and instead of focusing on specific devices we focus on the device classes (depending on the traffic characteristics they generate). In this way, a legitimate traffic profile is formed for each class of devices based on which DDoS detection models are developed. This approach has the potential to classify future devices based on the traffic flow characteristics they generate, which can be used to determine whether such a device behaves within legitimate limits or is generating DDoS traffic. Specifically, in our approach the DDoS traffic detection model is based on the logistic decision tree method from the set of machine learning methods. The problem of detecting DDoS traffic based on device classes has been reduced to binary classification, where different versions of the same model are developed for each class of SHIoT devices. This is why each class of SHIoT devices' traffic has different characteristics, which is evident from the presented versions of the model, each differing in the number of independent features used, the size of the decision tree and the threshold values of its branching. Our performance evaluation showed that the approach achieves high performance, in terms of accuracy, TPR, FPR, F1 rating, precision, ROC and PRC. For example, the accuracy of the model for respective classes is C1 = 99.9216%, C2 = 99.9966%, C3 = 99.9744%, and C4 = 99.9583%.

Our approach can benefit different stakeholders in the IoT ecosystem. For example, users typically want their devices to function as intended in the smart home environment. Generating DDoS traffic can impact on the device's functionality or make it completely inaccessible. Therefore, it is in the user's interest to promptly detect the device's abnormal behavior. Given that telecom operators are often also smart home service providers, it is also in their interest to detect unauthorized device behavior in a timely manner to protect their own network infrastructure. Finally, manufacturers of such devices must ensure the correct operation of the devices in order to increase customer satisfaction and strengthen their market presence.

While our research has demonstrated the potential of detecting illegitimate traffic with high accuracy based on the classification of devices into predefined classes and creating a legitimate traffic profile for each class using the boosting method of machine learning, there are a number of potential future extensions to this work. For example, we intend to evaluate our proposed approach in other settings, such as healthcare, transportation or Industry 4.0, as devices in these application domains may generate different behaviors and hence resulting in additional device classes. We also intend to study the potential of extending our approach to cover other attack types, for example to create device classes based on their generated traffic in the presence of other types of attacks.

REFERENCES

[1] I. Cvitić, D. Peraković, M. Periša, and S. Husnjak, "An overview of distributed denial of service traffic detection approaches," *PROMET Traffic Transp.*, vol. 31, no. 4, pp. 453–464, Aug. 2019, doi: 10.7307/ptt.v31i4.3082.

[2] G. A. Jaafar, S. M. Abdullah, and S. Ismail, "Review of recent detection methods for HTTP DDoS attack," *J. Comput. Netw. Commun.*, vol. 2019, pp. 1–10, Jan. 2019, doi: 10.1155/2019/1283472.

[3] AWS Shield Threat. (2020). *Threat Landscape Report—Q 1 2020 AWS*. Accessed: Oct. 29, 2020. [Online]. Available: https://aws-shield-tlr.s3.amazonaws.com/2020-Q1_AWS_Shield_TLR.pdf

[4] I. Cvitić, M. Vujić, and S. Husnjak, "Classification of security risks in the IoT environment," in *Proc. Ann. DAAAM Int. DAAAM Symp. Intell. Manuf. Autom.*, 2016, pp. 0731–0740, doi: 10.2507/26th.daaam.proceedings.102.

[5] I. Cvitić, D. Peraković, M. Periša, and M. Botica, "Smart home IoT traffic characteristics as a basis for DDoS traffic detection," in *Proc. 3rd EAI Int. Conf. Manage. Manuf. Syst.*, 2018, pp. 1–10, doi: 10.4108/eai.6-11-2018.2279336.

[6] I. Cvitić, D. Peraković, M. Periša, and B. Gupta, "Ensemble machine learning approach for classification of IoT devices in smart home," *Int. J. Mach. Learn. Cybern.*, to be published, doi: 10.1007/s13042-020-01241-0.

[7] P. M. S. Sánchez, J. M. J. Valero, A. H. Celdrán, G. Bovet, M. G. Pérez, and G. M. Pérez, "A survey on device behavior fingerprinting: Data sources, techniques, application scenarios, and datasets," Aug. 2020. [Online]. Available: http://arxiv.org/abs/2008.03343.

[8] R. Doshi, N. Apthorpe, and N. Feamster, "Machine learning DDoS detection for consumer Internet of Things devices," in *Proc. IEEE Security Privacy Workshops (SPW)*, San Francisco, CA, USA, May 2018, pp. 29–35, doi: 10.1109/SPW.2018.00013.

[9] Y. Meidan *et al.*, "N-BaIoT—Network-based detection of IoT Botnet attacks using deep autoencoders," *IEEE Pervasive Comput.*, vol. 17, no. 3, pp. 12–22, Jul.–Sep. 2018.

[10] M. Özçelik, N. Chalabianloo, and G. Gür, "Software-defined edge defense against IoT-based DDoS," in *Proc. 17th IEEE Int. Conf. Comput. Inf. Technol. (CIT)*, Helsinki, Finland, 2017, pp. 308–313, doi: 10.1109/CIT.2017.61.

[11] I. Cvitić, D. Peraković, M. Periša, and M. Botica, "Novel approach for detection of IoT generated DDoS traffic," *Wireless Netw.*, vol. 27, pp. 1573–1586, Jun. 2019, doi: 10.1007/s11276-019-02043-1.

[12] R. Hallman, J. Bryan, G. Palavicini, J. Divita, and J. Romero-Mariona, "IoDDoS—The Internet of distributed denial of service attacks—A case study of the Mirai malware and IoT-based botnets," in *Proc. 2nd Int. Conf. Internet Things Big Data Security*, 2017, pp. 47–58, doi: 10.5220/0006246600470058.

[13] D. H. Summerville, K. M. Zach, and Y. Chen, "Ultra-lightweight deep packet anomaly detection for Internet of Things devices," in *Proc. IEEE 34th Int. Perform. Comput. Commun. Conf. (IPCCC)*, Nanjing, China, Dec. 2015, pp. 1–8, doi: 10.1109/PCCC.2015.7410342.

[14] A. Saied, R. E. Overill, and T. Radzik, "Detection of known and unknown DDoS attacks using artificial neural networks," *Neurocomputing*, vol. 172, pp. 385–393, Jan. 2016, doi: 10.1016/j.neucom.2015.04.101.

[15] R. Vishwakarma and A. K. Jain, "A survey of DDoS attacking techniques and defence mechanisms in the IoT network," *Telecommun. Syst.*, vol. 73, no. 1, pp. 3–25, 2020, doi: 10.1007/s11235-019-00599-z.

[16] M. Saharkhizan, A. Azmoodeh, A. Dehghantanha, K.-K. R. Choo, and R. M. Parizi, "An ensemble of deep recurrent neural networks for detecting IoT cyber attacks using network traffic," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8852–8859, Sep. 2020, doi: 10.1109/jiot.2020.2996425.

[17] I. Ullah and Q. H. Mahmoud, "A two-level flow-based anomalous activity detection system for IoT networks," *Electronics*, vol. 9, no. 3, p. 530, Mar. 2020, doi: 10.3390/electronics9030530.

[18] O. Salman, I. H. Elhajj, A. Chehab, and A. Kayssi, "A machine learning based framework for IoT device identification and abnormal traffic detection," *Trans. Emerg. Telecommun. Technol.*, to be published, doi: 10.1002/ett.3743.

[19] E. Anthi, L. Williams, M. Slowińska, G. Theodorakopoulos, and P. Burnap, "A supervised intrusion detection system for smart home IoT devices," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 9042–9053, Oct. 2019, doi: 10.1109/JIOT.2019.2926365.

[20] D. Peraković, M. Periša, and I. Cvitić, "Analysis of the IoT impact on volume of DDoS attacks," in *Proc. 33rd Simpozijum o Novim Tehnologijama u poštanskom i Telekomunikacionom Saobraćaju (PosTel)*, 2015, pp. 295–304.

[21] N. Vlajic and D. Zhou, "IoT as a land of opportunity for DDoS hackers," *Computer*, vol. 51, no. 7, pp. 26–34, Jul. 2018, doi: 10.1109/MC.2018.3011046.

[22] N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac, and P. Faruki, "Network intrusion detection for IoT security based on learning techniques," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2671–2701, 3rd Quart., 2019, doi: 10.1109/COMST.2019.2896380.

[23] A. Sivanathan *et al.*, "Classifying IoT devices in smart environments using network traffic characteristics," *IEEE Trans. Mobile Comput.*, vol. 18, no. 8, pp. 1745–1759, Aug. 2019, doi: 10.1109/TMC.2018.2866249.

[24] I. Cvitic, D. Perakovic, M. Perisa, and M. Botica, *Definition of the IoT Device Classes Based on Network Traffic Flow Features* (EAI/Springer Innovations in Communication and Computing), L. Knapcikova, M. Balog, D. Perakovic, and M. Perisa, Eds. Cham, Switzerland: Springer, 2020, pp. 1–17.

[25] *GitHub—Markus-Go/Bonesi: BoNeSi—The DDoS Botnet Simulator*. Accessed: Aug. 7, 2019. [Online]. Available: https://github.com/Markus-Go/bonesi

[26] A. H. Lashkari, G. D. Gil, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of tor traffic using time based features," in *Proc. 3rd Int. Conf. Inf. Syst. Security Privacy (ICISSP)*, 2017, pp. 253–262, doi: doi: 10.5220/0006105602530262.

[27] N. Landwehr, M. Hall, and E. Frank, *Logistic Model Trees* (Lecture Notes in Artificial Intelligence (Lecture Notes in Computer Science 2837)). New York, NY, USA: Springer, 2003, pp. 241–252.

[28] N. Landwehr, M. Hall, and E. Frank, "Logistic model trees," *Mach. Learn.*, vol. 59, no. 1–2, pp. 161–205, 2005.

[29] B. Hssina, A. Merbouha, H. Ezzikouri, and M. Erritali, "A comparative study of decision tree ID3 and C4.5," *Int. J. Adv. Comput. Sci. Appl.*, vol. 4, no. 2, pp. 13–19, 2014, doi: 10.14569/specialissue.2014.040203.

[30] M. Hossin and M. Sulaiman, "A review on evaluation metrics for data classification evaluations," *Int. J. Data Min. Knowl. Manag. Process*, vol. 5, no. 2, pp. 01–11, Mar. 2015, doi: 10.5121/ijdkp.2015.5201.

[31] Y. Sasaki, "The truth of the F-measure," *Teach. Tuts. Mater.*, vol. 1, no. 4, pp. 1–6, 2007.

[32] S. Lang, F. Bravo-Marquez, C. Beckham, M. Hall, and E. Frank, "WekaDeeplearning4j: A deep learning package for Weka based on deeplearning4j," *Knowl. Based Syst.*, vol. 178, pp. 48–50, Aug. 2019, doi: 10.1016/j.knosys.2019.04.013.

[33] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1153–1176, 2nd Quart., 2016, doi: 10.1109/COMST.2015.2494502.

[34] S. J. Stolfo, W. Fan, W. Lee, A. Prodromidis, and P. K. Chan, "Cost-based modeling for fraud and intrusion detection: Results from the JAM project," in *Proc. DARPA Inf. Survivability Conf. Exposit. (DISCEX'00)*, vol. 2. Hilton Head, SC, USA, 2000, pp. 130–144, doi: 10.1109/DISCEX.2000.821515.

[35] *The CAIDA UCSD 'DDoS Attack 2007' Dataset*, CAIDA, La Jolla, CA, USA, 2007.

[36] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symp. Comput. Intell. Security Defense Appl.*, Ottawa, ON, Canada, Jul. 2009, pp. 1–6, doi: 10.1109/CISDA.2009.5356528.

[37] P. Gogoi, D. K. Bhattacharyya, B. Borah, and J. K. Kalita, "MLH-IDS: A multi-level hybrid intrusion detection method," *Comput. J.*, vol. 57, no. 4, pp. 602–623, 2014, doi: 10.1093/comjnl/bxt044.

[38] H. H. Jazi, H. Gonzalez, N. Stakhanova, and A. A. Ghorbani, "Detecting HTTP-based application layer DoS attacks on Web servers in the presence of sampling," *Comput. Netw.*, vol. 121, pp. 25–36, Jul. 2017, doi: 10.1016/j.comnet.2017.03.018.

[39] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. 4th Int. Conf. Inf. Syst. Security Privacy*, 2018, pp. 108–116, doi: 10.5220/0006639801080116.

[40] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset," *Future Gener. Comput. Syst.*, vol. 100, pp. 779–796, Nov. 2019, doi: 10.1016/j.future.2019.05.041.

[41] R. Paudel, T. Muncy, and W. Eberle, "Detecting DoS attack in smart home IoT devices using a graph-based approach," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, 2019, pp. 5249–5258, doi: 10.1109/BigData47090.2019.9006156.

[42] F. Hussain, S. G. Abbas, M. Husnain, U. U. Fayyaz, F. Shahzad, and G. A. Shah, "IoT DoS and DDoS attack detection using ResNet," in *Proc. IEEE 23rd Int. Multitopic Conf. (INMIC)*, Bahawalpur, Pakistan, Nov. 2020, pp. 1–6, doi: 10.1109/INMIC50486.2020.9318216.

[43] R. Paudel, T. Muncy, and W. Eberle, "Detecting DoS attack in smart home IoT devices using a graph-based approach," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Los Angeles, CA, USA, Dec. 2019, pp. 5249–5258, doi: 10.1109/BigData47090.2019.9006156.

[44] Y. Meidan, V. Sachidananda, H. Peng, R. Sagron, Y. Elovici, and A. Shabtai, "A novel approach for detecting vulnerable IoT devices connected behind a home NAT," *Comput. Security*, vol. 97, Oct. 2020, Art. no. 101968, doi: 10.1016/j.cose.2020.101968.

Ivan Cvitić received the master's degree from the Faculty of Transport and Traffic Sciences and the Ph.D. degree in the field of technical sciences from University of Zagreb, Zagreb, Croatia, in 2013 and 2020, respectively.

He is currently with the Faculty of Transport and Traffic Sciences, University of Zagreb, as a Postdoctoral Researcher and an Associate with the Laboratory for Security and Forensic Analysis of Information and Communication System. He has published more than 50 scientific papers at international conferences, scientific books, and highly rated scientific journals. His research domain and interests are in cybersecurity, applied machine learning and artificial intelligence, modeling network traffic anomalies, DDoS, Internet of Things, digital forensics, and communication networks.

Dr. Cvitić is a member of the editorial board, reviewer board, and a guest editor for several highly rated scientific journals and international conferences.

Dragan Peraković (Member, IEEE) received the master's and Ph.D. degrees in the field of technical sciences from the Faculty of Transport and Traffic Sciences (FPZ), University of Zagreb, Zagreb, Croatia, in 2003 and 2005, respectively.

After graduation, he began his career with FPZ, where he is currently working as a Full Professor and holds the positions of Head of Department for Information and Communication Traffic and Head of Chair of Information and Communication Systems and Services Management. He has engaged in several international scientific projects and Research and Development studies as a researcher, a leading researcher, and an evaluator. Also, he has authored or coauthored of more than 150 scientific papers and a member, board member, and official editor of several journals and conferences in his research field. His current research interest is in security, digital forensic, innovative communication services in the transport system, smart city, and industry 4.0.

Brij B. Gupta (Senior Member, IEEE) received the Ph.D. degree in the area of Information and cyber security from the Indian Institute of Technology Roorkee, Roorkee, India, in 2011.

He is currently working as Assistant Professor with the Department of Computer Engineering, National Institute of Technology Kurukshetra, Kurukshetra, India. He is also working as a Principal Investigator of various Research and Development projects. He was also a Visiting Researcher with Yamaguchi University, Yamaguchi, Japan, in 2015; Deakin University, Geelong, VIC, Australia, in 2017; and Swinburne University of Technology, Melbourne, VIC, Australia in 2018. Additionally, he was a Visiting Professor with Temple University, Philadelphia, PA, USA, June, 2019, and Staffordshire University, Stoke-on-Trent, U.K., in July 2019. He published more than 300 research papers in International Journals and Conferences of high repute. His research interests include information security, cyber security, cloud computing, Web security, intrusion detection, and Phishing.

Kim-Kwang Raymond Choo (Senior Member, IEEE) received the Ph.D. degree in information security from the Queensland University of Technology, Brisbane, QLD, Australia, in 2006.

He currently holds the Cloud Technology Endowed Professorship with The University of Texas at San Antonio, San Antonio, TX, USA.

Prof. Choo is a recipient of the 2019 IEEE Technical Committee on Scalable Computing Award for Excellence in Scalable Computing (Middle Career Researcher), the 2018 UTSA College of Business Col. Jean Piccione and Lt. Col. Philip Piccione Endowed Research Award for Tenured Faculty, the Outstanding Associate Editor of 2018 for IEEE ACCESS, the British Computer Society's 2019 Wilkes Award Runner-Up, the 2014 Highly Commended Award by the Australia New Zealand Policing Advisory Agency, the Fulbright Scholarship in 2009, the 2008 Australia Day Achievement Medallion, and the British Computer Society's Wilkes Award in 2008. He has also received Best Paper Awards from the IEEE SYSTEMS JOURNAL in 2021, *IEEE Consumer Electronics Magazine* for 2020, *EURASIP Journal on Wireless Communications and Networking* in 2019, IEEE TrustCom 2018, and ESORICS 2015; the Korea Information Processing Society's *Journal of Information Processing Systems* Outstanding Research Award (Most-Cited Paper) for 2020 and Survey Paper Award (Gold) in 2019; the IEEE Blockchain 2019 Outstanding Paper Award; and the Best Student Paper Awards from Inscrypt 2019 and ACISP 2005. He is named the Cybersecurity Educator of the Year—APAC (Cybersecurity Excellence Awards are produced in cooperation with the Information Security Community on LinkedIn) in 2016, and in 2015, he and his team won the Digital Forensics Research Challenge organized by Germany's University of Erlangen-Nuremberg. He is the Founding Chair of IEEE Technology and Engineering Management Society's Technical Committee on Blockchain and Distributed Ledger Technologies, and serves as the Department Editor of IEEE TRANSACTIONS ON ENGINEERING MANAGEMENT, and the Associate Editor of IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, and IEEE TRANSACTIONS ON BIG DATA. He is an ACM Distinguished Speaker and an IEEE Computer Society Distinguished Visitor from 2021 to 2023, and included in Web of Science's Highly Cited Researcher in the field of Cross-Field in 2020.