Novel Datasets and Traffic Generation Tools for Intrusion Detection in IoT Communications

Giovanni Stanco, Stefania Zinno, Pietro Violante, Alessio Botta, Giorgio Ventre
DIETI Department, *University of Napoli Federico II*, Naples, Italy
{giovanni.stanco, stefania.zinno, alessio.botta, giorgio.ventre}@unina.it, pi.violante@studenti.unina.it

***Abstract*—Datasets constitute the foundation for the training of a robust Intrusion Detection System (IDS). In particular, IDSs could be useful in IoT scenarios, in which resource-constrained devices can not run traditional security software and require a monitoring entity within their network. In this research, we move along two main directions: on one hand, the analysis and description of significant datasets regarding the security of IoT communications; on the other, the exploration of tools that enable their generation. Following this division, we first depict IoT datasets available in the literature and compare them with respect to their most important characteristics according to the literature. We also compare them according to the most popular features that are typically collected in such scenarios, including *network and transport layer information*, *temporal parameters*, *payload characteristics*, and *statistical indicators*. We then move our attention to the software tools and frameworks which enable the generation of benign or malicious network traffic in various experimental environments and are used in the construction of datasets. We believe this work could serve as a useful starting point to develop novel approaches and techniques in the field.**

***Index Terms*—IoT networks, IoT communications, IoT security, Datasets**

I. Introduction

The Internet of Things (IoT) brought a revolution to the technological landscape, introducing new possibilities of interaction among objects, environments, and people [1]. The growing number of connections of intelligent devices in various sectors, including home automation, industry, healthcare, and public infrastructures, has enabled numerous benefits in terms of automation, efficiency, and accessibility. The limited resources available to IoT devices do not enable the adoption of strong security and monitoring protocols, as in the case of other complex network scenarios [2], resulting in exposure to a wide variety of risks, including significant challenges regarding communication security [3], [4].

Due to the increasing number of security risks and complexity of malicious techniques, the ability to promptly detect anomalous malicious behavior through monitoring and Intrusion Detection Systems (IDSs) becomes fundamental. In order to build data-driven systems able to protect an IoT system, it is important to study and analyze the data coming from various security threats [5]. To ensure the effectiveness of IDS solutions, it is fundamental to have representative datasets that include both real and emulated traffic, with accurate labels and well-defined attack scenarios. The quality of a dataset, along with the quality of the results researchers could obtain from its analysis, strongly depends on the tools used to generate it. Various tools are currently available for the generation of such datasets, including network frameworks, custom scripts, and virtualized test environments.

This work provides an updated survey of the main datasets regarding the security of IoT communications and of the tools that can be employed for generating such datasets. The goal is to provide a clear, technical, and up-to-date overview of the tools available to the community working in the field of IoT communication security.

After an overview on current literature about this topic (Sec. II), we introduce the main concerns for the security of IoT communications (Sec. III) and present the most popular datasets used in research within this field, highlighting their main characteristics (Sec. IV). We then cover the tools employed for the generation of relevant datasets and modern solutions that currently serve as benchmarks in research and experimentation (Sec. V). Conclusions are drawn in Sec. VI.

II. Related Works

Over the years, many researchers have focused on the vulnerabilities of IoT devices and the available data for security analysis, ultimately leading to the development and evaluation of IDSs. As of 2020, there was still no single comprehensive source addressing the vulnerabilities affecting IoT devices [6]. In 2021, a systematic review was carried out to understand the nature of data collected by the IoT devices [7]. In 2023, a security review analyzed 74 datasets, categorizing them according to their focus on IoT devices, benign or malicious network traffic, and their availability [8]. A more comprehensive study provided a summary of Intrusion Detection research in IoT networks conducted between 2018 and 2024, including information on datasets, attack types, experimental results, and classifiers used in the studies [9]. Other examples of review works only focus on a specific dataset [10]. Concerning the tools, performance comparison of software as PackETH, Ostinato, and D-ITG in terms of bandwidth utilization over the link has been investigated [11]. Dataset generation still represents one of the biggest challenges in this field. With the aim of providing a key resource for IoT security, innovative tools are able to generate synthetic data for balanced datasets [12].

Our work combines both a study on datasets and tools, is fully up to date as of 2025, and provides a comprehensive taxonomy to support dataset selection.

978-3-903176-75-1 ©2025 IFIP

III. Context and Motivation

A. Context

The evolution of wireless communication technologies has fostered the emergence of IoT [13]. IoT devices, equipped with sensors, lightweight processors, and network interfaces, are able to sense the environment around them to collect data that are transmitted to and processed by powerful computing entities. The value of the information that could be retrieved from the gathered data exposes such devices to an increasing number of security threats. Because of the intrinsic features of IoT devices, including limited power supply and limited computational power, and of poor user practices such as use of default passwords, security in these systems is hindered.
To give context, we provide a brief introduction to the many possible threats that IoT systems may be vulnerable to. First of all, attacks on IoT systems may intervene directly in the system, modifying the data flow, injecting malicious code, or blocking resources, as in Distributed Denial of Service (DDoS) attacks [14]. Another goal of threats in these systems may be to collect data for later analysis and future attacks.

The security of IoT solutions may be compromised at *(i) physical level*, causing unauthorized access to devices, forced disconnection, and physical damage; *(ii) network level*, by injecting packets or carrying out MAC spoofing and Man-in-the-Middle (MitM) attacks; or *(iii) application level*, exploiting application protocol vulnerabilities to access data or services. Approaches to carry out such attacks include: *(i) Malicious software* that compromises IoT devices and the corporate networks they belong to; *(ii) Cryptanalysis*, since the limited resources of IoT devices leave no choice but to adopt lightweight cryptographic algorithms vulnerable to attacks; *(iii) Resource exhaustion*, saturating resources through anomalous traffic generated by bots, and rendering services unavailable; *(iiii) Web attacks*, for example, Denial-of-Service (DoS) attacks against web servers or sniffing of sensitive data.

B. Motivation

Traditional security systems, such as antivirus software and firewalls, prove inadequate in the IoT context due to the heterogeneity of the devices and to the limited computing resources on which such systems should run. In such a scenario, IDSs become fundamental for protecting communications [15]. In recent years, the self-learning capabilities of Artificial Intelligence (AI) have shown great promise in enhancing intrusion detection in IoT systems, particularly for detecting evolving threats. Thanks to the AI capabilities, IoT systems are able to defend themselves by learning from previous attacks, improving their ability to detect and respond to a wide variety of attacks. With the advancement of AI, Machine Learning (ML) and Deep Learning (DL) approaches have become widespread for intrusion detection. In this scenario, the role of datasets becomes central: they represent the empirical foundation on which to build, train, and evaluate intrusion detection models in IoT environments. A well-designed IDS requires realistic and diverse data to learn how to distinguish between legitimate and malicious traffic. Data can be collected either in emulated environments, where conditions and attacks can be controlled, or in real-world scenarios, which better represent the natural behavior of devices and the complexity of the network. Datasets have become essential benchmarks for the development of IDSs, and should provide comprehensive and representative data to effectively respond to the diverse and continuously evolving security threats. In this regard, the quality, availability, and representativeness of the dataset play a fundamental role in assessing the practical effectiveness of the system.

IV. Datasets

A. Analysis of state-of-the-art datasets

We now introduce the main datasets used as a reference for security analysis in IoT communications. We provide a brief summary of their content and scope, while we discuss their characteristics in the following subsections.

N-BaIoT *[16]:* dataset acquired within a controlled laboratory, emulating a real-world scenario of a home network populated by commercial IoT devices. This configuration reflects the typical context of botnet attacks, which primarily target local networks connecting common everyday devices.

BoT-IoT *[17]:* dataset generated in a realistic virtualized environment, replicating a smart home network connecting devices including refrigerators, smart lights, thermostats, etc. Realistic communication scenarios in domestic IoT environments, which are vulnerable to botnet attacks, are reproduced.

Kitsune *[18]:* this network attack dataset, introduced alongside the Kitsune network intrusion detection system, includes both benign traffic and various malicious scenarios (e.g., ARP spoofing, DoS) in a realistic IoT setting. The dataset was collected in a realistic IoT environment, using a Raspberry Pi acting as the gateway.

MQTTset *[19]:* MQTTset was created to represent communications based on the Message Queue Telemetry Transport (MQTT) protocol, which is widely used in IoT applications. Considered scenarios are smart home or smart building environments, comprising various sensors, such as temperature, humidity, motion, gas, smoke, and locks, all identified by an IP address and connected to an MQTT broker.

TON_IoT *[20]:* dataset containing Telemetry, Operating systems, and Network data collected from a testbed representing a realistic medium-scale network. The testbed is designed to replicate real-world scenarios of smart homes, smart cities, or industrial environments, comprising physical and virtual end-devices, Edge-Fog-Cloud infrastructures, MQTT services, and vulnerable Web services. Telemetry data were collected from IoT and Industrial IoT (IIoT) sensors.

Edge-IIoTset *[21]:* this dataset was gathered to represent realistic IoT and IIoT scenarios, such as agricultural, urban, and healthcare environments. The testbed is composed of seven layers (Cloud, NFV, blockchain, Fog, SDN, Edge, perception) and reproduces a complex cyber-physical ecosystem with smart devices and distributed servers. It is considered one of the most comprehensive for IoT/IIoT system security.

TABLE I: Comparison of the datasets.

Dataset	Domain	No. of Devices	Traffic Type	Network Type and Components	Data/Format	Country	Benign	Malicious
N-BaIoT [16]	Home	9	Real	Wi-Fi	Pcap, features	Negev, IL	50,295	6,506,674
BoT-IoT [17]	Home	5	Virtual	Firewall, router, gateway, MQTT server	Pcap, features	Canberra, AU	9,543	73,360,900
Kitsune [18]	Home, Surveillance	9	Real	Wi-Fi	Pcap, features	Negev, IL	700,000	26,472,754
MQTTset [19]	Home	10	Virtual	MQTT broker	Pcap, features	Genoa, IT	11,915,716	165,463
TON_IoT [20]	Home, City Industry	7	Virtual	VM, MQTT broker, web server	Pcap, features, logs	Canberra, AU	245,000	124,619
Edge-IIoTset [21]	Industry	10	Virtual	Edge server, MQTT broker, Modbus TCP, Cloud server, blockchain	Pcap, features	Guelma, DZ	11,223,940	9,728,708

B. Dataset characteristics

The first comparison among the mentioned datasets for IDS training for IoT networks was conducted following the key properties identified in the literature, listed in the following.

- **Domain**: datasets may focus on different applications, including home environments, industrial settings, etc.;
- **Number of IoT devices**: a larger number of involved devices ensures representativeness of the dataset;
- **Traffic Type**: real, when acquired from actual devices operating in real environments, or virtual, when generated by software tools in controlled environments;
- **Collection Duration**: the time window of data recording (e.g., days, weeks), useful for estimating traffic dynamics;
- **Network Type and Components**: wireless technology, LAN, cellular network, etc., indicating whether the dataset is suitable for specific application contexts;
- **Availability**: available only upon request or publicly accessible, enabling reproducibility of studies;
- **Data Format**: datasets can be provided in PCAP format (raw network packets with payload), log format (system or server events), or feature-level format (values already processed for ML training);
- **Labeling**: each dataset instance can be labeled as benign, malicious, or multiclass (e.g., type of attack);
- **Country of origin**: the collection location may influence network behavior (e.g., regional configurations, languages, ISPs);
- **Number of Samples (Benign/Malicious)**: number of labeled instances (packets, flows, or events) in each dataset, divided into benign and malicious traffic. A large and balanced number of samples in a dataset is preferable for training reliable models.

A detailed comparison of the characteristics of the datasets is reported in Table I. Due to space constraints, we omit the columns for Availability (all the datasets are public), Labeling (all the datasets have labeled samples), and Duration (only for MQTTset we know that the collection lasted for a week). From the comparison, we can notice that all the datasets involved a limited number of devices, which generated a very large number of samples in all the cases, except TON_IoT. On the other hand, **TON_IoT** is the only one providing the event logs, and the one focusing on the widest variety of scenarios.

C. Main features in datasets

To properly exploit the data gathered in these datasets, it is important to be aware of the features that can be found in these data structures. Knowing which information has been collected is important to plan a certain analysis and run it to achieve significant results. For this reason, we compare the mentioned datasets according to the most popular features that are typically collected in such scenarios.

The most common and representative features found in IoT traffic datasets include network and transport layer information, temporal parameters, payload characteristics, and statistical indicators. In particular:

1) **Source IP address**: useful for identifying the origin of the traffic and detecting compromised devices or spoofing;
2) **Destination IP address**: used to reconstruct flows, connections, and attack patterns;
3) **Source port**: which may be fixed or dynamic, legitimate or anomalous;
4) **Destination port**: indicating the target service;
5) **Protocol**: indicating distinct traffic types;
6) **Timestamp/Duration**: time indication to detect fast or persistent attacks;
7) **TCP Flags**: state of the TCP connections, for identifying scans, incomplete connections, or floods.
8) **Packet length**: anomalous traffic may have very small or very large size in bytes;
9) **Inter Arrival Time (IAT)**: time elapsed between two consecutive packets, used to identify bursts, floods, or suspicious delays;
10) **Raw payload (PCAP)**: actual content of the packet for deep analysis;
11) **Statistical indicators**: calculated on packets or flows, fundamental for ML classifiers;
12) **MQTT-specific features**: specific information for the MQTT protocol (message type, topic, QoS, etc.);
13) **Traffic label**: information for supervised learning: binary or multiclass field indicating whether the traffic is benign or malicious or indication of the type of attack.

Table II shows which of these features are present in the main datasets analyzed. As observed in Table II, the fundamental features for describing network traffic — such as the source

TABLE II: Presence of the main features in the analyzed IoT datasets. The first column reports the citation for each dataset, the rest of the columns refer to the features as listed in Sec. IV-C.

	Features												
	1	2	3	4	5	6	7	8	9	10	11	12	13
[16]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓
[17]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓
[18]	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	✗	✗
[19]	✗	✗	✗	✗	✓	✗	✗	✓	✓	✓	✓	✓	✓
[20]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
[21]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

IP address and the destination port — are not always present in all the datasets. Less common features include protocol-specific ones, such as MQTT metrics or detailed labels, which are available only in certain cases. The most complete datasets, in terms of coverage and variety of features, are **Edge-IIoTset** and **TON_IoT**, while **MQTTset** presents the most limited set of attributes.

V. Software Tools for Data Generation

In this section, we focus on the frameworks and software tools used in the construction of the datasets, starting from the generation of benign or malicious network traffic in various experimental environments.

Node-RED: Node-RED is an open-source programming tool for building IoT applications based on logical communication flows between sensors, cloud services, and user interfaces [22]. It is a suitable solution for orchestrating realistic IoT scenarios, generating benign or malicious traffic (e.g., through anomalous MQTT messages, high-frequency flows, or malicious commands to vulnerable devices), triggering unexpected events, and controlling virtual or physical devices connected via Wi-Fi. Node-RED is typically installed on edge devices, such as Raspberry Pi, capable of acquiring data from sensors and sending them via the MQTT protocol to centralized cloud services collection systems, realizing a complete edge-to-cloud IoT architecture [23]. Node-RED played a key role in the generation of the *BoT-IoT* and *TON_IoT* datasets.

MQTT Security Assistant (MQTTSA): MQTTSA is a tool designed to automatically detect vulnerabilities in MQTT brokers and provide detailed suggestions for their mitigation [24]. Thanks to its traffic analysis capabilities, it is possible to detect fuzzing, DoS attacks, and brute-force attempts. Moreover, MQTTSA allows to generate a report describing the detected vulnerabilities, correlating them with practical exploit examples and secure configuration recommendations. MQTTSA modules have been used to recreate various attack scenarios, including data exfiltration (unauthorized transmission of sensitive information through legitimate MQTT topics) and DoS attacks through intensive publishing or malformed packets [25]. This tool was used in the generation of the *MQTTset* and *Edge-IIoTset* datasets.

MQTT-malaria: MQTT-malaria is a targeted tool used for generating specific malicious traffic over the MQTT protocol [26]. This tool has been used to replicate fuzzing attacks and send high-frequency MQTT packets with unstructured payloads, with the aim of testing the resilience of the MQTT broker and training IDSs [27]. This tool enabled the construction of datasets containing both benign and malicious packets, including *MQTTset* and *Edge-IIoTset* datasets.

IoT-Flock: IoT-Flock is an open-source framework capable of emulating both legitimate and malicious IoT devices in real environments [28]. It supports two of the most common application protocols in this domain: MQTT and CoAP. IoT-Flock can generate four main known attacks on the MQTT protocol, including MQTT Publish Flood and MQTT Packet Crafting Attack. *MQTTset* and *Edge-IIoTset* are two examples of datasets produced thanks to the use of this emulator.

Ostinato: Ostinato is an open-source tool for network traffic generation, designed to create customized packet flows in order to test and analyze network behavior [29]. It allows both the generation and manipulation of packets, supporting a wide range of network protocols of different layers. Additionally, it integrates Wireshark directly into the working environment, allowing importing PCAP files and facilitating the verification of generated traffic and the testing activities. Ostinato's architecture is based on two components able to run on separate machines: the controller, serving as the user interface, and the drone, responsible for traffic transmission and packet capture. *BoT-IoT* is among the datasets generated using this tool.

Simulation using Pyhton and Java scripts: In the context of dataset generation for security analysis in IoT networks, another common method consists of using scripts developed in Python or JavaScript for controlled simulation of benign and malicious actors. This technique is particularly useful for both generating benign traffic (e.g., MQTT, HTTP, CoAP communications) and reproducing attacks (e.g., injection, MitM, DDoS). A notable example of this methodology is the proposal of the Meta Attack Language (MAL), a formal language designed to model attack scenarios through graph construction [30]. Specifications written in MAL can be compiled into programming languages, enabling automatic simulations of compromise within specific domains. The resulting scripts can model causal sequences of attacks with a level of abstraction that allows repeatability and scalability of the simulation experiment. Similarly, Kudzu is a symbolic execution framework for automatic client-side code analysis [31]. The system enables automatic generation of execution paths that include XSS and injection vulnerabilities, highlighting how JavaScript can be used as a means of simulating input behavior and attacks in web services. Thanks to the flexibility of scripts, it is possible to define sensor logic (e.g., thresholds, periodic events), temporal sequences of malicious actions, and packet structure (using libraries such as Scapy or Pyshark in Python). In many real data collections (e.g., BoT-IoT, Edge-IIoTset), these scripts are often accompanied by PCAP files and automatically generated CSV annotations, thus representing a standardized and programmable method for producing labeled traffic.

Kali Linux: Kali Linux is a Debian-based distribution designed for security analysis [32]. It is widely adopted in

academic and professional contexts to test system robustness, thanks to the availability of hundreds of pre-installed tools for vulnerability analysis, packet sniffing, and wireless network attacks. The many tools for testing activities make it a flexible and customizable framework [33].

In our context of interest, Kali Linux is used to generate realistic malicious traffic by mimicking attacks such as brute force, spoofing, and MitM. These scenarios can be reproduced through the combined use of tools included in the system.

Kali Linux was used in the generation of *BoT-IoT*.

SLICES: Scientific Large-scale Infrastructure for Computing/Communication Experimental Studies (SLICES) is a European research infrastructure designed to enable advanced experimentation in IoT, 5G/6G networks, cloud-edge computing, and communication security [34]. It is a federated platform, distributed across multiple European countries, that allows the creation of realistic and reproducible test environments for large-scale experiments.

The infrastructure enables the configuration of physical and virtual devices, the definition of legitimate or malicious traffic scenarios, and data collection through advanced monitoring tools. This makes it particularly suitable for generating datasets aimed at evaluating security solutions such as IDS systems or behavioral classification algorithms. A distinctive aspect of SLICES is the adoption of a *reproducibility-by-design* methodology, which integrates tools and processes to ensure the repeatability and replicability of experiments. This approach is based on live Linux images and full automation of experimental workflows, facilitating the reproduction of results by other researchers and promoting collaborative innovation and the evolution of networking technologies.

6G Sandbox Toolkit: The 6G Sandbox Toolkit is a set of experimental tools developed within the European 6G-SANDBOX project, funded by the Horizon Europe initiative [35]. This toolkit is designed to support experimentation and evaluation of emerging technologies for future 6G networks, including advanced IoT solutions, edge computing, and approaches focused on network security and resilience. The goal of the toolkit is to provide a unified framework for the deployment, orchestration, and monitoring of distributed testbeds, leveraging open-source tools and standardized interfaces. It includes modules for network traffic generation, edge node management, and automated data collection, facilitating the creation of realistic datasets useful for training ML models for intrusion detection or behavioral classification. The 6G Sandbox Toolkit offers native integration with virtualized and containerized environments, making it compatible with cloud-native infrastructures. This makes it particularly useful for recreating and validating heterogeneous IoT scenarios, with the ability to automate large-scale security tests and attacks.

Kathará: Kathará is an open-source network emulation system based on containers, designed to simplify the creation and management of complex network scenarios [36]. It uses Docker containers to emulate network devices, linked via virtual LANs, allowing the emulation of realistic networks on single machines or distributed clusters. This tool is particularly useful for interactive demonstrations and testing production networks in sandbox environments. Kathará supports integration with tools such as Quagga, Open vSwitch, and P4, offering flexibility in configuring emulated network devices.

MATLAB: MATLAB, together with Simulink, offers a comprehensive platform for the design and deployment of IoT applications. It is widely used to model connected devices, generate realistic network traffic, and analyze data from sensors or virtual environments [37]. One of MATLAB's distinctive features is its integration with ThingSpeak, a cloud-based IoT platform that enables the transmission and reception of data from virtual sensors, facilitating the creation of realistic scenarios for evaluating their security. Moreover, MATLAB supports IoT traffic generation using statistical models and stochastic processes, for example using Poisson processes or constant bit rate models. Interestingly, MATLAB offers tools for generating Narrowband IoT (NB-IoT) waveforms, both in uplink and downlink, in compliance with 3GPP specifications and analyzing network performance in controlled scenarios [38].

VI. Conclusion

The growing spread of IoT devices has highlighted the need to strengthen the security of communications between interconnected systems. The complex and dynamic scenarios in which these devices operate, often in critical and distributed environments, pose new challenges in terms of protection against attacks and anomaly detection.

In this work, an analysis of the main datasets available for IoT network security and of the software tools used for network traffic generation is presented. It clearly emerged that the quality of a dataset depends not only on the variety and quantity of collected data, but above all on the ability to replicate realistic, reproducible, and well-documented scenarios. Software tools, therefore, are fundamental components in the process of building datasets useful for the development and validation of intrusion detection systems. This study analyzed both emulators and programming tools actually used in real datasets, such as Node-RED and IoT-Flock, and next-generation infrastructures like SLICES and 6G Sandbox Toolkit. Although the latter are not always integrated into currently available datasets, they represent a valuable resource for future experimentation, thanks to their flexibility and adherence to realistic scenarios.

Having a good knowledge of these tools not only enables the generation of high-quality data but also allows for the design of accurate testbeds and the replication of experiments in different contexts. In a field where scientific reproducibility and result comparability are key elements, having a solid infrastructure for generating and collecting data is essential.

In conclusion, security in IoT communications cannot disregard the availability of reliable data and the conscious use of traffic generation tools. This work, through an approach focused on the analysis of both datasets and underlying tools, aims to offer a useful reference for those involved in designing,

testing, or improving security solutions in complex and ever-evolving IoT environments.

ACKNOWLEDGMENT

This work was partially supported by project cyberHuman, part of the SERICS program (PE00000014) under the MUR National Recovery and Resilience Plan funded by the European Union - NextGenerationEU, and by the European Union - Next Generation EU, Mission 4, Component 1, through the ADAPTO project, part of the RESTART program, CUP E63C2 2002040007, CP PE0000001.

REFERENCES

[1] Ala Al-Fuqaha, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aledhari, and Moussa Ayyash. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Communications Surveys & Tutorials*, 17(4):2347–2376, 2015.

[2] Alessio Botta, Roberto Canonico, Annalisa Navarro, Giovanni Stanco, and Giorgio Ventre. Towards a highly-available sd-wan: Rapid failover based on bfd protocol. In *2023 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, pages 153–158, 2023.

[3] Antonia Affinito, Stefania Zinno, Giovanni Stanco, Alessio Botta, and Giorgio Ventre. The evolution of Mirai botnet scans over a six-year period. *Journal of Information Security and Applications*, 79:103629, 2023.

[4] Giovanni Stanco, Annalisa Navarro, Flavio Frattini, Giorgio Ventre, and Alessio Botta. A comprehensive survey on the security of low power wide area networks for the Internet of Things. *ICT Express*, 10(3):519–552, 2024.

[5] Iqbal H Sarker, Asif Irshad Khan, Yoosef B Abushark, and Fawaz Alsolami. Internet of things (IoT) security intelligence: a comprehensive overview, machine learning solutions and research directions. *Mobile Networks and Applications*, 28(1):296–312, 2023.

[6] Marcin Rytel, Anna Felkner, and Marek Janiszewski. Towards a Safer Internet of Things—A Survey of IoT Vulnerability Data Sources. *Sensors*, 20(21), 2020.

[7] Ritika Lohiya and Ankit Thakkar. Application domains, evaluation data sets, and research challenges of IoT: A systematic review. *IEEE Internet of Things Journal*, 8(11):8774–8798, 2021.

[8] François De Keersmaeker, Yinan Cao, Gorby Kabasele Ndonda, and Ramin Sadre. A survey of public IoT datasets for network security research. *IEEE Communications Surveys & Tutorials*, 25(3):1808–1840, 2023.

[9] Sulyman Age Abdulkareem, Chuan Heng Foh, Mohammad Shojafar, François Carrez, and Klaus Moessner. Network intrusion detection: An IoT and Non IoT-related survey. *IEEE Access*, 12:147167–147191, 2024.

[10] Jared M. Peterson, Joffrey L. Leevy, and Taghi M. Khoshgoftaar. A review and analysis of the Bot-IoT dataset. In *2021 IEEE International Conference on Service-Oriented System Engineering (SOSE)*, pages 20–27, 2021.

[11] Shalvi Srivastava, Sweta Anmulwar, AM Sapkal, Tarun Batra, Anil Kumar Gupta, and Vinodh Kumar. Comparative study of various traffic generator tools. In *2014 Recent Advances in Engineering and Computational Sciences (RAECS)*, pages 1–6. IEEE, 2014.

[12] Miada Almasre and Alanoud Subahi. Create a realistic IoT dataset using conditional generative adversarial network. *Journal of Sensor and Actuator Networks*, 13(5), 2024.

[13] Subhas Chandra Mukhopadhyay and Nagender K Suryadevara. Internet of Things: Challenges and opportunities. *Internet of Things: Challenges and opportunities*, pages 1–17, 2014.

[14] Stefania Zinno, Giovanni Di Stasi, Stefano Avallone, and Giorgio Ventre. A load balancing algorithm against DDoS attacks in beyond 3G wireless networks. In *2014 Euro Med Telco Conference (EMTC)*, pages 1–6, 2014.

[15] Md Mahbubur Rahman, Shaharia Al Shakil, and Mizanur Rahman Mustakim. A survey on intrusion detection system in IoT networks. *Cyber Security and Applications*, 3:100082, 2025.

[16] Yair Meidan, Michael Bohadana, Yael Mathov, Yisroel Mirsky, Asaf Shabtai, Dominik Breitenbacher, and Yuval Elovici. N-BaIoT—Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders. *IEEE Pervasive Computing*, 17(3):12–22, 2018.

[17] Nickolaos Koroniotis, Nour Moustafa, Elena Sitnikova, and Benjamin Turnbull. Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. *Future Generation Computer Systems*, 100:779–796, 2019.

[18] Yisroel Mirsky, Tomer Doitshman, Yuval Elovici, and Asaf Shabtai. Kitsune: An ensemble of autoencoders for online network intrusion detection. In *The Network and Distributed System Security Symposium (NDSS) 2018*, 2018.

[19] Ivan Vaccari, Giovanni Chiola, Maurizio Aiello, Maurizio Mongelli, and Enrico Cambiaso. MQTTset, a new dataset for machine learning techniques on MQTT. *Sensors*, 20(22):6578, 2020.

[20] Abdullah Alsaedi, Nour Moustafa, Zahir Tari, Abdun Mahmood, and Adnan Anwar. TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems. *Ieee Access*, 8:165130–165150, 2020.

[21] Mohamed Amine Ferrag, Othmane Friha, Djallel Hamouda, Leandros Maglaras, and Helge Janicke. Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning. *IEEe Access*, 10:40281–40306, 2022.

[22] IBM Emerging Technology. Node-RED. https://nodered.org/, 2013. Accessed: 2025-06-07.

[23] Milica Lekić and Gordana Gardašević. IoT sensor integration to Node-RED platform. In *2018 17th International Symposium Infoteh-Jahorina (Infoteh)*, pages 1–5. IEEE, 2018.

[24] A. Palmieri, P. Prem, S. Ranise, U. Morelli, and T. Ahmad. MQTT Security Assistant (MQTTSA). https://sites.google.com/fbk.eu/mqttsa, 2019. Accessed: 2025-06-08.

[25] Andrea Palmieri, Paolo Prem, Silvio Ranise, Umberto Morelli, and Tahir Ahmad. MQTTSA: A tool for automatically assisting the secure deployments of MQTT brokers. In *2019 IEEE World Congress on Services (SERVICES)*, volume 2642, pages 47–53. IEEE, 2019.

[26] etactica. mqtt-malaria: scalabilità e strumenti di stress per MQTT. https://github.com/etactica/mqtt-malaria, 2015. Accessed: 2025-06-08.

[27] Ege Ciklabakkal, Ataberk Donmez, Mert Erdemir, Emre Suren, Mert Kaan Yilmaz, and Pelin Angin. ARTEMIS: An intrusion detection system for MQTT attacks in Internet of Things. In *2019 38th Symposium on Reliable Distributed Systems (SRDS)*, pages 369–3692. IEEE, 2019.

[28] Syed Ghazanfar, Faisal Hussain, Atiq Ur Rehman, Ubaid U Fayyaz, Farrukh Shahzad, and Ghalib A Shah. IoT-Flock: An Open-source Framework for IoT Traffic Generation. In *2020 International Conference on Emerging Trends in Smart Technologies (ICETST)*, pages 1–6. IEEE, 2020.

[29] Srivats P. Ostinato: Packet/Traffic Generator and Analyzer. https://ostinato.org/, 2023. Accessed: 2025-08-26.

[30] Pontus Johnson, Robert Lagerström, and Mathias Ekstedt. A meta language for threat modeling and attack simulations. In *Proceedings of the 13th international conference on availability, reliability and security*, pages 1–8, 2018.

[31] Prateek Saxena, Devdatta Akhawe, Steve Hanna, Feng Mao, Stephen McCamant, and Dawn Song. A symbolic execution framework for JavaScript. In *2010 IEEE Symposium on Security and Privacy*, pages 513–528. IEEE, 2010.

[32] Offensive Security. Kali Linux: Penetration Testing and Ethical Hacking Linux Distribution. https://www.kali.org/, 2023. Accessed: 2025-08-26.

[33] Renas R Asaad. Penetration testing: Wireless network attacks method on Kali Linux OS. *Academic Journal of Nawroz University*, 10(1):7–12, 2021.

[34] SLICES-RI Project. SLICES Research Infrastructure. https://www.slices-ri.eu/, 2024. Accessed: 2025-08-26.

[35] 6G SANDBOX Project. 6G Sandbox Toolkit. https://6g-sandbox.eu/6g-sandbox-toolkit/, 2024. Accessed: 2025-08-26.

[36] Kathará Project. Kathará: Lightweight container-based network emulation system. https://www.kathara.org/, 2024. Accessed: 2025-05-27.

[37] MathWorks. Internet of Things - MATLAB & Simulink. https://www.mathworks.com/solutions/internet-of-things.html, 2024. Accessed: 2025-08-26.

[38] MathWorks. NB-IoT Uplink Waveform Generation. https://www.mathworks.com/help/lte/ug/nb-iot-uplink-waveform-generation.html, 2024. Accessed: 2025-08-26.