Advancing Cybersecurity in IoT: A Data-Driven Approach to Discovering Unknown Botnet Attacks

Innocent Mbona[1,*], Jan H. P. Eloff[1]
[1]Department of Computer Science, University of Pretoria, South Africa
Emails: u15256422@tuks.co.za; jan.eloff@up.ac.za

Abstract

Over the years, exciting new technologies such as the Internet of Things (IoT) have changed many aspects of our lives, including smart homes. Unfortunately, this technology is vulnerable to cyber attacks owing to the lack of physical boundaries to ensure safety, privacy, and security. Botnet attacks are among the prominent cybersecurity threats because they can compromise the entire network with cyber attacks, such as distributed denial-of-service (DDoS) attacks. Hence, the intelligent discovery of new unknown botnet attacks remains a challenge, particularly in IoT environments, owing to the complex nature of the signatures of unknown botnet attacks. Through a systematic literature review, we provide a comprehensive review of current studies to determine the trends and challenges in the discovery of unknown botnet attacks. This study implemented a lightweight intelligent data-driven methodology called CySecML to discover unknown botnet attacks. The CySecML methodology differs from existing methods because of its unique data preparation and feature selection methods, specifically aimed at mitigating cyber attacks. The effectiveness of this methodology is demonstrated using state-of-the-art botnet attack data sets, where the self-training machine-learning algorithm achieved the best results with an F_1-score of 94%.

Keywords: Botnet; Internet of Things (IoT); Unknown attacks; Cybersecurity; Feature selection; Machine learning; Network intrusion detection system

1. Introduction

Over the past ten years, the proliferation of interconnected devices has accelerated significantly, largely driven by advancements in Internet of Things (IoT) technology [1]. IoT enhances global intelligence by effortlessly linking a wide range of devices - from compact personal gadgets like portable Wi-Fi routers to large-scale systems such as autonomous vehicles [1]. Recent studies such as those by Saadouni et al. [2] demonstrated that IoT applications are rapidly growing worldwide, and are expected to generate billions of revenues in the next five years [2]. IoT technology enables wireless communication and interaction among numerous devices, facilitating a substantial exchange of data across the connected network [2]. While this technology is undoubtedly exciting, it also presents significant cybersecurity challenges, including concerns around data privacy and network protection [1]. IoT devices are susceptible to cyber threats, such as distributed denial-of-service (DDoS) attacks, due to the absence of physical boundaries that typically safeguard security, privacy, and safety [1]. Consequently, much of the current research in IoT is centred around developing robust cybersecurity measures to detect and prevent serious threats, such as DDoS attacks orchestrated by botnets - a network of compromised devices controlled by a central botmaster. Unlike bot attacks, which are launched using a single bot, botnet attacks are coordinated attacks launched by a botmaster. Therefore, discovering a network of malicious bots differs from discovering a malicious bot because each bot within a network may display its own unique behaviour. In addition, the techniques used for generating synthetic botnets differ from those used for bot attacks, such as their data sets. Botnet attacks are challenging to detect because cybercriminals continuously exploit emerging or previously unknown network vulnerabilities, known as zero-day vulnerabilities. These threats can be mitigated using a Network Intrusion Detection System (NIDS) - a strategically placed software within a network. When powered by machine learning

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

or deep learning, NIDS becomes a smart solution, leveraging predictive and classification algorithms to identify abnormal behaviours, such as botnet attacks.

Botmasters control their bots either through centralised architecture such as command and control (C&C) or decentralised architecture such as peer-to-peer (P2P). A centralised architecture allows a botmaster to communicate directly with each bot in a network through a C&C server, whereas in a decentralised architecture, each bot acts as a server. Botnet attacks generally occur in three phases. First, a botmaster identifies a vulnerability in a network system or antivirus software and exploits this vulnerability, that is, the initial stage of the attack. Second, a botmaster can infect a victim's machine with malware and distribute the attack to control the machine fully. This stage is considered infected and distributed. Finally, the botmaster executes the attacks and expands them to multiple devices via malicious e-mails and HTTP links. A well-known case study of a botnet attack in the IoT environment is the Mirai botnet attack that occurred in 2016. Since the Mirai botnet attack, researchers and practitioners have proposed various methods for discovering and preventing this constantly increasing cybersecurity threat. It is crucial to discover unknown cyber-attacks timeously so that the vulnerability can be patched and possibly recover any lost data. Figure 1 illustrates the C&C architecture (top part), which is the focus of this study, and P2P architecture (bottom part).

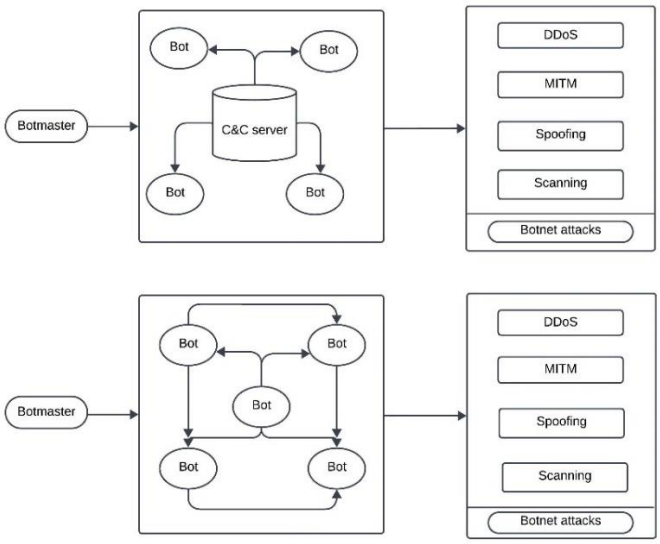

Figure 1. Botnet attack types

Lately, considerable research has been conducted to enhance the discovery pillar of cybersecurity frameworks, particularly for botnet attacks. State-of-the-art research highlights that machine learning (ML) based cybersecurity solutions can effectively and intelligently discover botnet attacks, especially within large-scale data environments such as IoT. Therefore, the quality of data is a crucial factor to be considered when designing ML based cybersecurity solutions because poor data can lead to unreliable ML model outcomes. In addition, there is a need to enhance existing ML based cybersecurity solutions to be lightweight so that they can cope with voluminous data generated from big data platforms such as IoT. Existing cybersecurity methodologies lack a metric approach for assessing data quality and computationally efficient methods for identifying complex network traffic such as unknown botnet attacks. This study proposes the CySecML methodology to address these challenges. The below research questions are addressed in this study.

1.1. Research questions

(i) What are the prominent cyber threats posed by botnets in IoT?

(ii) What are the current trends and challenges in discovering botnet attacks?

(iii) Which features are significant for the tracing and intelligent discovery of unknown botnet attacks?

(iv) How can machine learning-based cybersecurity solutions be enhanced to effectively discover unknown centralised botnet attacks?

2. Related Work

To address the research questions formulated in the previous section, we first developed a search query to identify state-of-the-art papers relevant to the study at hand, to intelligently discover unknowns, that is, zero-day botnet attacks in IoT. The search query: ("zero-day" OR "unknown") AND ("botnet" OR "bot-net") AND ("IoT") AND ("machine learning") was applied to Scopus, DBLP Computer Science Bibliography, IEEE Xplore Digital

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

Library, ACM Digital Library , SpringerLink, Web of Science and Google Scholar databases to identify papers published over the past five years (2019 to 2024) written in English. Using the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) framework, the following studies were identified.

As shown in Table 1, 129 studies were identified based on the search query described above. The PRISMA framework was adopted to identify the most relevant papers. The PRISMA framework provides a structured approach to identifying relevant papers to address research question(s) by defining the identification, screening, eligibility, and inclusion steps. The PRISMA steps followed in this study are outlined in Figure 2, whereby 23 papers were identified as relevant to the study.

Table 1: Literature review - identified studies

Database	**Papers identified**	**No full access**	**Final papers**
Scopus	22	4	18
IEEE Xplore Digital Library	16	0	16
SpringerLink	38	6	32
Web of Science	29	6	23
ACM Digital Library	10	0	10
DBLP Computer Science Bibliography	4	0	4
Google Scholar	26	0	26
Total	**145**	**16**	**129**

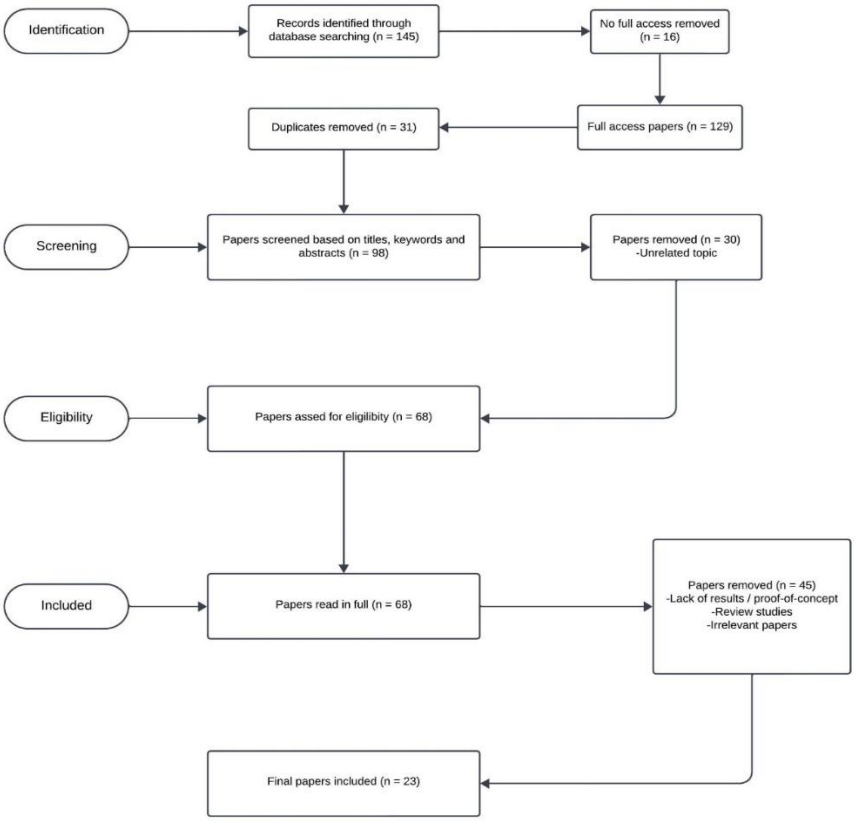

Figure 2. PRISMA framework steps

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

2.1. Literature review – discovering unknown botnet attacks in IoT

This section discusses the studies identified in Figure 2 following the PRISMA framework. This was performed to determine trends and evaluate the strengths and limitations of state-of-the-art methods in discovering unknown botnet attacks. In addition, the data sets used to train the ML algorithms to discover botnet attacks are indicated.

Liu et al. [3] introduced deep learning (DL) algorithms aimed at discovering malicious network traffic within IoT environments, particularly unknown attacks, given the vast data volumes generated by IoT devices. Their approach involved using an embedded model to extract meaningful features for input into the DL algorithms. Specifically, they employed random forest for feature selection in combination with a convolutional neural network (RCNN), and used XGBoost alongside a convolutional neural network (XCNN) to identify IoT-based attacks. Additionally, they developed a data set named the Center for Cyber Defense IoT Network Intrusion Data set V1 (CCD-INID-V1), which reflects real-world attack scenarios. This data set includes various attack types such as ARP poisoning, SlowLoris, Hydra brute-force targeting Asterisk, UDP flood, and ARP denial-of-service (DoS), all simulated in a virtual smart-home environment using IoT devices and sensors. However, the paper does not clearly specify the botnet configuration used to generate the data set, which led to its exclusion from this study. Among the models tested, the RCNN achieved the highest accuracy at 96%.

Zhang et al. [4] introduced a federated learning (FL) approach for discovering unknown botnet attacks in IoT networks. Unlike traditional machine learning, FL is designed for decentralized systems, where each IoT device functions as an independent server. This approach was chosen due to the study's focus on peer-to-peer (P2P) botnets, which operate in a decentralized manner. The authors evaluated their method using the N-BaIoT data set and achieved an accuracy of 80%. However, since this study concentrates on centralized botnets, as outlined in the Introduction, the FL-based approach was not adopted here.

Ohtani et al. [5] proposed an intrusion detection approach known as IDAC (Intrusion Detection based on Attack Candidates) for identifying unknown botnet attacks in IoT networks. The IDAC method targets centralized botnet systems using one-class support vector machines (OCSVM), and decentralized systems through federated learning (FL). Their evaluation utilized the BoT-IoT data set, which includes various attack types such as DDoS, data exfiltration, OSScan, ServiceScan, and keylogging. To extract relevant features, they applied principal component analysis (PCA), although this technique can be computationally intensive on large-scale IoT platforms, potentially making the system resource-heavy. The IDAC model achieved an F_1-score of 97% for centralized systems and 99% for decentralized systems. Similarly, Ohtani et al. [5] employed FL and OCSVM to discover botnet attacks in IoT networks, using the N-BaIoT data set, and reported an F_1-score of approximately 99%.

Alkahtani et al. [6] proposed a hybrid deep learning model combining convolutional neural networks (CNN) and long short-term memory (LSTM), referred to as CNN-LSTM, for discovering unknown botnet attacks in IoT networks. Their approach focused on centralized botnet detection using the N-BaIoT data set, which includes Mirai and Bashlite attacks. The model achieved an accuracy of 91% by utilizing all 115 features from the data set as input. However, using the full feature set without prior selection poses challenges: (i) redundant features may degrade the performance of deep learning models, especially in identifying unknown threats, and (ii) it can significantly increase computational overhead, making the model less efficient for large-scale IoT environments.

Kumar et al. [7] developed a two-phase framework for botnet discovering, consisting of signature generation and attack detection. In the first phase, graph-based techniques were used to generate signatures for known attacks and their variants. These signatures were then utilized in the detection phase to identify botnet activity. A key limitation of this approach is its inability to discover unknown attacks, as their signatures are not predefined. To evaluate the framework, the authors generated synthetic real-time attacks - including DoS/DDoS, probing, data exfiltration, and keylogging - and categorized a subset as known attacks for signature generation, while the rest were treated as unknown. Additionally, the CICIDS18 data set was used for validation, and the framework achieved an accuracy of approximately 91.6% on this data set.

Krishnan et al. [8] developed a methodology comprising two main components: an analysis module and a detection module, designed to identify both known and unknown attacks in IoT networks. The analysis module involves collecting network traffic data, normalizing it, and selecting relevant features using Localized Generalized Matrix Learning Vector Quantization (LGMLVG). This technique identifies important features through a relevance matrix constructed using the Fisher discriminant ratio. However, a notable limitation of this approach is the computational complexity involved in calculating covariance matrices for high-dimensional data sets. Once key features are selected, the detection module employs a class-wise stacked multi-classifier to discover

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

attacks. The methodology was evaluated using the ToN_IoT-20 and CICIDS18 data sets, achieving F_1-scores of 93% and 97%, respectively.

Celdran et al. [9] proposed a methodology that combines behavioural fingerprinting with machine learning (ML) to discover unknown botnet attacks in IoT networks. Behavioural fingerprinting involves profiling a device based on its activity patterns, such as the nature of data it transmits. Devices compromised by malware typically exhibit behavioural deviations from normal, non-infected devices. However, this technique can be challenging when applied to environments with numerous devices and standardized security protocols, which may obscure unique behavioural traits. The extracted behavioural data is then fed into ML algorithms for attack detection. The authors employed both supervised and semi-supervised learning approaches, achieving F_1-scores of 94% with XGBoost and 96% with a one-class semi-supervised model. The attack data used for evaluation was synthetically generated.

Abdalgawad et al. [11] explored the use of deep learning algorithms - specifically adversarial autoencoders (AAE) and bidirectional generative adversarial networks (BiGAN) - to discover unknown botnet attacks in IoT networks. Their study utilized the IoT-23 data set, which includes various attack types such as Mirai botnet activity. Instead of applying traditional feature selection techniques, the authors removed highly correlated features to enhance the efficiency and effectiveness of the deep learning models. Among the approaches tested, BiGAN delivered the highest performance, achieving an F_1-score of 85% in identifying unknown botnet attacks.

Popoola et al. [12] proposed a federated deep learning (FDL) approach for discovering unknown botnet attacks in IoT networks. FDL was chosen to preserve data privacy, as it operates within a decentralized system. The authors employed a deep neural network (DNN) to identify botnet activity and evaluated their method using the Bot-IoT and N-BaIoT data sets. To simulate real-world scenarios involving unknown attacks, they split the attack data between training and testing phases. Their FDL approach achieved an F_1-score exceeding 85%. However, since this study focuses on centralized botnets, federated learning falls outside its scope.

Ibrahim et al. [13] assessed the effectiveness of convolutional neural networks (CNNs) in discovering unknown botnet attacks within IoT networks. To address overfitting issues, they introduced various regularization techniques. Their evaluation was based on the Bot-IoT data set, which includes a range of botnet attacks such as DDoS and keylogging. With regularization applied, the CNN model achieved an accuracy exceeding 91%. Using the information gain (IG) feature selection method, the authors identified features like “sequence number,” “number of transmitted packets,” and “duration” as particularly important for intelligent botnet detection. However, a key limitation of their study is the reliance on a single data set, which restricts the generalizability of their findings - especially regarding the significance of selected features for tracing botnet activity.

Khraisat et al. [14] proposed an ensemble-based hybrid intrusion detection system combining a one-class support vector machine (OCSVM) with a C5 classifier to identify botnet attacks in IoT networks. The C5 classifier, which uses a binary decision tree approach, initially classifies network traffic as either benign or malicious. Traffic that remains unclassified by the C5 model is then passed to the OCSVM, which is trained exclusively on benign samples to accurately learn normal behaviour. During testing, any deviation from this learned benign behaviour is flagged as anomalous, indicating potential unknown attacks. The authors evaluated their method using the Bot-IoT data set and achieved an accuracy exceeding 93%. They also applied the information gain (IG) feature selection method, similar to the approach by Ibrahim et al. [13], but identified 13 significant features compared to Ibrahim et al [13]. This discrepancy underscores the need for further research to refine feature selection for effective botnet detection.

Alharbi et al. [15] introduced a graph-based machine learning approach for discovering botnet attacks in IoT networks. Their methodology leverages graph techniques to uncover botnet behavioural patterns, representing significant features as edges within a graph structure. To identify these features, the authors applied five filter-based feature selection methods, including information gain (IG). The approach was evaluated using the CTU-13 and IoT-23 data sets, achieving an F_1-score of 50% and a precision of 34% in discovering unknown botnet attacks. These results highlight the inherent challenges in accurately identifying unknown botnet threats.

Ahmad et al. [16] proposed a methodology that integrates mutual information with deep learning (DL) algorithms, such as deep neural networks (DNN), to discover botnet attacks in IoT networks. Mutual information is used to reduce uncertainty by quantifying the shared information between features and data classes (benign or malicious), making it a valuable tool for feature selection. The authors applied this technique to identify significant features and evaluated various DL models using the IoT-Botnet 2020 data set. Their approach achieved an accuracy of approximately 99%. However, a key limitation of the study is its reliance on a single data set, which may restrict the generalizability of the results.

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

Roshan et al. [17] proposed an optimized autoencoder-based deep learning approach for discovering unknown botnet attacks. Their method was evaluated using the CICIDS2017 data set. To enhance model efficiency, they applied a correlation-based feature selection technique, removing highly correlated features prior to training. Optimization of the autoencoder was achieved by tuning input parameters, including varying the detection threshold (10%, 15%, 20%, and 25%). The approach achieved an accuracy of 99% in identifying botnet attacks. Notably, their study demonstrated how detection thresholds influence the performance of ML/DL models, with lower thresholds (10% or 15%) yielding higher accuracy. However, a limitation of their work is the reliance on a single data set, which restricts the generalizability of their findings across different ML/DL models and botnet data sets.

Li et al. [18] introduced a framework named ADRIoT, designed to discover unknown botnet attacks in IoT networks using an anomaly detection approach. The framework consists of three main components: network traffic capture, data preprocessing, and an anomaly detection module. For the detection phase, the authors employed an unsupervised LSTM autoencoder, chosen for its ability to learn from unlabelled data and thus effectively handle unknown threats. However, unsupervised methods are known to suffer from high false positive and false-negative rates, particularly when the model struggles to distinguish between benign and malicious traffic. The authors evaluated their approach using custom-generated data, including DDoS and scanning attacks, and reported an area under the curve (AUC) of approximately 98%.

Luu et al. [21] introduced a deep learning model called the Deep Sparse Contrastive Autoencoder (DSCAE) for detecting both known and unknown botnet attacks in IoT networks. Their study utilized the N-BaIoT data set, which includes botnet attacks such as Mirai and Gafgyt. In addition to DSCAE, the authors implemented various machine-learning algorithms, including random forest, and applied principal component analysis (PCA) for feature selection. While PCA is effective, it can be computationally intensive - particularly on large-scale IoT platforms - potentially making the detection model resource-heavy. The DSCAE model achieved an F_1-score of approximately 94% in identifying unknown botnet attacks.

Liang et al. [23] introduced the use of Long Short-Term Memory (LSTM) networks to discover previously unknown Denial-of-Service (DoS) botnet attacks. This approach was also adopted by Alkahtani et al. [6] and Li et al. [18] for botnet detection. A key strength of LSTM lies in its ability to distinguish between benign and malicious network traffic using only a limited number of packet flows. The researchers evaluated their method using the CICIDS2017 data set. Furthermore, they compared the performance of the LSTM model with other machine learning algorithms, including neural networks and support vector machines. The LSTM model achieved an F_1-score of 80%. However, a notable limitation of the study was its reliance on a single data set for validation.

Alazzam et al. [24] explored the application of supervised machine learning algorithms to discover botnet attacks targeting IoT networks. Their study employed Random Forest (RF), Naïve Bayes (NB), and k-Nearest Neighbour (kNN) classifiers. The rationale behind using supervised learning lies in the high volume of data generated by IoT environments, which makes it feasible to obtain labelled samples. However, a key drawback of this approach is that unknown botnet attacks lack identifiable signatures, leading to potential mislabelling issues. The researchers utilized the N-BaIoT data set and applied information gain for feature selection. Among the tested algorithms, RF demonstrated the highest performance, achieving an F_1-score of approximately 99%.

Blaise et al. [26] introduced a technique known as split-and-merge, which leverages an unsupervised, port-based machine learning approach to discover botnet attacks. The core idea of their method is to monitor network ports for anomalous behaviour - such as a sudden surge in traffic directed at a specific port - which may indicate malicious activity. Unlike supervised methods, the unsupervised algorithm identifies attacks without relying on labelled training data. The researchers evaluated their approach using the MAWI and UCSD network telescope data sets and employed the F-test for feature selection. Their methodology achieved a true positive rate (TPR) of approximately 86%. Nonetheless, the challenges associated with using unsupervised techniques to discover unknown attacks remain a topic of ongoing discussion.

Li et al. [27] proposed a hybrid semi-supervised machine learning approach that combines signature-based (supervised) and anomaly-based (unsupervised) techniques to discover denial-of-service (DoS) botnet attacks in IoT networks. The motivation behind this approach is to harness the strengths of both methods - using algorithms like Random Forest for signature-based detection and LSTM for anomaly-based detection - to improve attack identification. The researchers evaluated their methodology using the NSL-KDD, BoT-IoT, and CICIDS2018 data sets, achieving an F_1-score of approximately 98% in identifying unknown botnet attacks. Building on this work, our study broadens the scope by incorporating a wider variety of botnet attacks, including man-in-the-middle (MITM), flooding, and scanning attacks, among others.

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

Nitish et al. [28] proposed an online intrusion detection framework utilizing the Extreme Learning Machine (EXL) to identify unknown botnet attacks. Their approach addresses the issue of class imbalance, a common challenge in IoT networks where benign traffic significantly outweighs malicious traffic. The framework was evaluated using the Bot-IoT and NSL-KDD data sets, achieving an impressive F_1-score of approximately 99%. EXL is a feedforward neural network that operates under supervised learning principles, with the unique characteristic that weights assigned to hidden layers remain fixed after initial assignment. Despite its effectiveness, the limitations of supervised machine learning in detecting previously unseen attacks have been discussed earlier.

Khan et al. [30] introduced a lightweight deep learning framework that combines artificial neural networks (ANN) with self-organizing maps to discover botnet attacks in IoT environments. Self-organizing maps, which operate under an unsupervised learning paradigm, cluster data based on the distance between nodes, as previously discussed. The authors evaluated their approach using the NSL-KDD and N-BaIoT data sets, achieving an overall accuracy of approximately 84%.

The studies discussed above are summarized in Table 2 and used later to benchmark the proposed methodology for discovering unknown botnet attacks.

Table 2: Summary of state-of-the-art approaches in discovering unknown botnet attacks in IoT

Author(s)	**Year**	**Botnet type**	**Data set(s)**	**Data quality checks**	**Feature selection method**	**Discovery approach**
Liu et al. [3]	2021	C&C	CCD-INID-V1 Balot DoH20	×	Random forest XGBoost	Deep learning
Zhang et al. [4]	2023	P2P	N-BaIoT	×	Not specified	Federated learning
Ohtani et al. [5]	2024	P2P and C&C	BoT-IoT	×	PCA	Federated learning and OCSVM
Alkahtani et al. [6]	2021	C&C	N-BaIoT	×	Not applied	Hybrid deep learning
Kumar et al. [7]	2021	C&C	CICIDS18	×	Not applied	Graph based approach
Krishnan et al. [8]	2024	C&C	NF-BoT-IoT NF-ToN-IoT NF-CSE-CIC-IDS2018	×	LGMLVG	Two-Level Shallow Autoencoder
Celdran et al. [9]	2022	C&C	Generated by authors	×	Not specified	Behavioural fingerprinting and machine learning
Abdalgawad et al. [11]	2022	C&C	IoT-23	×	Correlation measure	Deep learning
Popoola et al. [12]	2022	P2P	Bot-IoT N-BaIoT	×	Not applied	Federated deep learning
Ibrahim et al. [13]	2023	C&C	Bot-IoT	×	Information gain	Deep learning

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

Khraisat et al. [14]	2019	C&C	Bot-IoT	×	Information gain	Hybrid machine learning
Alharbi et al. [15]	2021	C&C	IoT-23 CTU-13	×	Correlation measure	Graph based machine learning
Ahmad et al. [16]	2021	C&C	IoTID-20	×	Mutual information	Deep learning
Roshan et al. [17]	2021	C&C	CICIDS2017	×	Correlation measure	Deep learning
Li et al. [18]	2022	C&C	Their own data	×	Not specified	Deep learning
Luu et al. [21]	2023	C&C	N-BaIoT	×	Not applied	Deep learning
Liang et al. [23]	2019	C&C	CICIDS2017	×	Not applied	Deep learning
Alazzam et al. [24]	2019	C&C	N-BaIoT	×	Information gain	Machine learning
Blaise et al. [26]	2020	C&C	MAWI UCSD network telescope	×	F-Test score	Machine learning
Li et al. [27]	2024	C&C	NSL-KDD BoT-IoT CICIDS2018	×	Information gain Fast correlation based filter	Hybrid Deep learning
Nitish et al. [28]	2024	C&C	Bot-IoT NSL-KDD	×	Correlation measure	Extreme learning machine
Khan et al. [30]	2023	C&C	NSL-KDD N-BaIoT	×	Not specified	Deep learning
Our paper	2025	C&C	IoT-23 IoTID-20 TON_IoT-20 N-BaIoT	✓	Benford's law	Machine learning

2.1.1. Our contribution

Our contributions were based on the research questions formulated in this study. Specifically, the CySecML methodology is proposed to intelligently discover botnet attacks in IoT networks. This methodology addresses the following research gaps: (i) how to quantitatively perform data quality checks in IoT data sets, (ii) propose a computationally efficient feature selection method for tracing botnet attacks using Benford's law, and (iii) demonstrate that this methodology is effective and lightweight compared to existing methodologies. CySecML is described in the following section.

3. Methodology for discovering IoT botnet attacks

The CySecML methodology is developed to provide organizations with guidelines for developing machine-learning-based cybersecurity solutions. Specifically, the CySecML methodology provides a structured approach

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

that can guide organizations to ensure that important aspects, including extracting cybersecurity data sets, checking data quality, and feature selection, are adequately implemented, particularly for discovering complex attacks such as unknown botnet attacks. The CySecML methodology is depicted in Figure 3.

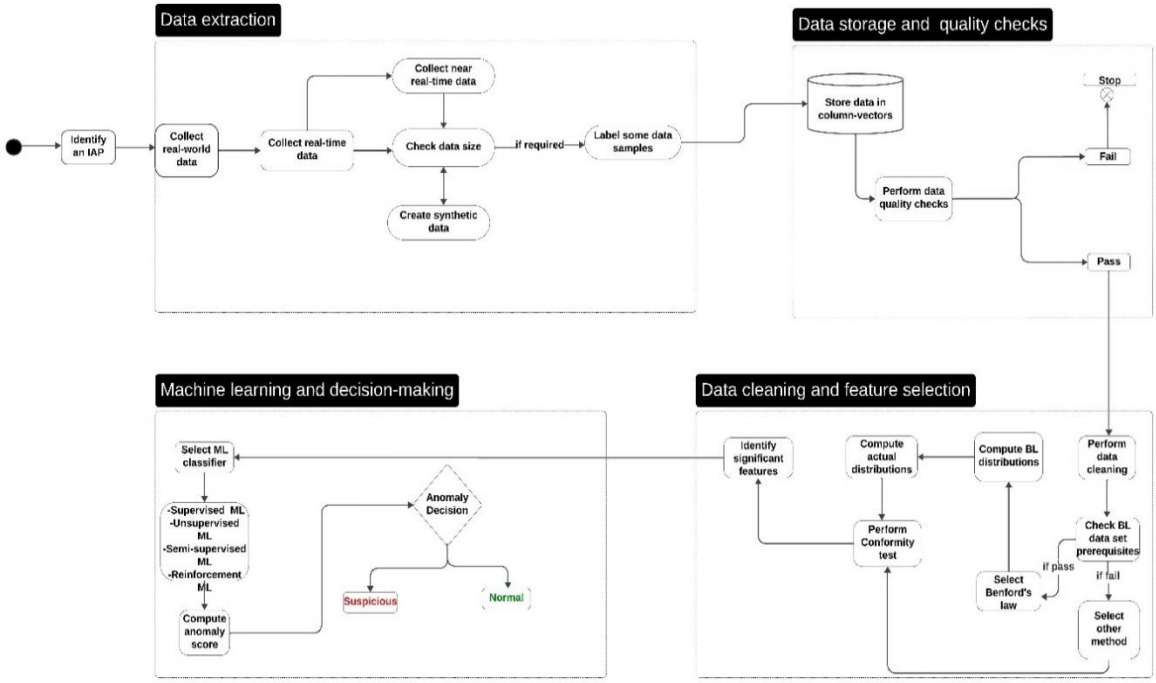

Figure 3. The CySecML methodology

The CySecML methodology is applicable to Internet application platforms (IAPs), such as IoT, to intelligently discover cyber-attacks. This methodology is unique from other existing cybersecurity methodologies, given its unique components of data extraction, data quality checks, and Benford's law properties to identify significant features in cybersecurity data sets. The details of the CySecML methodology steps are provided below.

Identify an IAP: The first step is to identify a suitable IAP where CySecML will be implemented; for instance, IoT is an IAP used in this study. People and organizations form a part of cyberspace elements; therefore, the protection of humans from cybercrime is of crucial importance from a cybersecurity perspective.

Collect real-world data: This step involves collecting real-world data from an IAP that is used to discover cyber attacks. ML based cybersecurity solutions should be trained and tested on a real-world data set such that the solutions used can produce trusted results.

Collect real-time data: This step relates to instantaneous user data collected from an IAP. State-of-the-art data-driven cybersecurity solutions depend on real-time data to effectively discover cyber attacks. Data such as attributes and features are used to describe users. For example, the Destination IP address" feature describes the IP address of the server to which network traffic is forwarded.

Collect near real-time data: This step deals with collecting near real-time user data because collecting real-time data can be challenging in some instances. For example, network connectivity issues may affect real-time data availability.

Check data size: This step determines the size of the collected data. ML algorithms should ideally be trained on a sample size ten times larger than the number of features or attributes. For example, if a data set contains five features, then the reasonable sample size should be at least 50. For example, the IoTID-20 data set consists of 82 features with data samples of over 620000. If the collected real-world data are insufficient, as per the above, then an alternative would be to supplement it with synthetic data.

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

Create synthetic data: If an organization were not in possession of sufficient real-world data, then an alternative would be to consider obtaining synthetic attack data. Various tools exist, such as the CICFlowmeter that can be used to generate network-traffic data. ML based and FL based cybersecurity solutions rely on cybersecurity data availability.

Label some data samples: This step involves assigning labels to data samples, and its approach varies depending on the selected machine learning algorithm: supervised learning requires fully labelled data; unsupervised learning operates without labelled data; semi-supervised learning uses a mix of labelled and unlabelled data; and reinforcement learning relies on feedback rather than labelled data.

Store data in column vectors: This step stores the collected data. For ML algorithms, the data set is preferred to be stored in column vectors, mainly to simplify the implementation of feature selection.

Perform data quality checks: This step aims to ensure that the data sets used to train the ML based cybersecurity solutions are of good quality and suitable for cybersecurity problems. ML based cybersecurity solutions should be trained and tested on trusted and reliable data sets to produce dependable results. If a cybersecurity data set fails to meet all these data quality checks except for presentation quality, one should not proceed further, because training a cybersecurity ML algorithm using unreliable data will result in unreliable outcomes that can lead to high false alarm rates.

Perform data cleaning: This step involves cleaning the collected data. A cybersecurity analyst or the system controller of the IAP typically completes this step. Cybersecurity data from an IAP are not always clean enough to build ML based cybersecurity solutions. For instance, cybersecurity data may contain corrupt or invalid information, such as blank fields, on the features that may occur during data processing. Moreover, in this study, negative values were removed so that Benford's law (BL) could be applied to real positive numbers as a feature selection method.

Check BL data set prerequisites: This step deals with the minimum data set prerequisites that must be met to correctly use BL. Not all cybersecurity data sets from IAPs are expected to conform to BL; therefore, an assessment must be conducted to determine whether a cybersecurity data set meets the BL data set prerequisites. For example, one of the prerequisites for the BL data set is that the leading digits, 1 to 9, must be observable. If all BL prerequisites are satisfied, BL can be selected as a feature selection method; otherwise, a different feature selection method, such as information gain can be selected.

Compute BL distributions: This step deals with the computation of BL distributions using BL properties. BL distributions form part of the building blocks for identifying significant features that are indicative of anomalous behaviours in cybersecurity data sets.

Compute actual distributions: This step deals with the computation of the cleaned (actual) distributions of the leading digits following the previous block.

Conformity test: This step compares the BL with the actual distribution of the leading digits. Statistical methods, such as the Pearson chi-square distribution test, were conducted at a 95% confidence level.

Identify significant features: The importance of identifying significant features prior to implementing ML based cybersecurity solutions is crucial and well understood in the cybersecurity community, as this can ultimately impact the overall performance and computational cost of a cybersecurity ML algorithm.

Compute anomaly score: This step forms part of the ML algorithm component and computes an anomaly score based on the precision, recall, F_1-score, and MCC score.

Anomaly decision: This is the output of the ML algorithm, where a decision is made regarding whether a specific botnet activity is malicious or normal. For illustrative purposes, in this study, if an activity has either an MCC or an F_1-score exceeding 80%, it is deemed malicious; however, it is considered normal if these scores are below the 50% threshold. A self-adaptable threshold can be considered if system controllers choose this approach. Having described the components of the CySecML methodology, the next section provides a pseudocode representation of the CySecML methodology to illustrate its implementation.

3.1 Pseudocode

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

The blocks involved in CySecML methodology are discussed above. However, these blocks do not provide guidelines for real-world implementation of this methodology. To fill this void, a pseudocode representation is presented as follows:

Block I – data extraction
Select an IAP and its users $U = \{u_1, u_2, \dots u_n\}$.
Let $F = \{f_1, f_2, \dots f_m\}$ be a set of features from the IAP that describe the activities of U. Collect real-time data D from an IAP, check the data size, and add synthetic data if required.
Block II – data storage and quality checks
if D passes data quality checks, then
perform data cleaning
else
stop the process
end
D is a union of X_L and X_U where
$X_L = \{x_1, x_2, \dots, x_m\}$ is a set of labelled data points of set U.
$X_U = \{x_1, x_2, \dots, x_m\}$ is a set of unlabelled data points of set U.
Let S be a feature selection method and let Benford's law (BL) in this case.
Block III – data cleaning and feature selection
if D passes BL data set prerequisites then
select BL as a feature selection method
else
select other method
end
For all $f_i \in F$ compute BL distributions on set X_L
If f_i obeys one of the BL distributions on a normal data set while violating all the BL distributions on a malicious data set, then
add $f_i$to a subset $F' \subseteq F$
else
disregard f_i
end
Block IV – machine learning and decision-making
Let $h(\Theta)$ be a machine learning classifier
(i) For all $U \in X_U$, label X_U using $h(X_L|F')$
(ii) Compute the anomaly score $\beta \in [0,100]$ using $h(X_L|F')$
if $\beta \geq 80\%$ then
label u_i as a malicious behaviour
else
label u_i as a benign behaviour
end

The CySecML methodology is implemented next to provide proof-of-concept.

4. Experimentation and results

section outlines the experiments conducted to validate the CySecML methodology. The initial phase of the CySecML framework - data extraction is discussed below.

Data extraction experiments
We conducted a high-level assessment of the state-of-the-art data sets used to train ML based cybersecurity solutions to discover botnet attacks. A summary of these data sets is provided in Table 2. We then selected four data sets: IoT-23, IoTID-20, TON_IoT-20, and N-BaIoT. These data sets were selected because they contain the most recent types of botnet attacks in IoT networks. The IoT-23 data set is presented below.

IoT-23 data set
The IoT-23 data set was developed in 2020 by the Avast AIC Laboratory and Czech Technical University in Prague. This data set consists of three IoT devices: a smart door lock, Philips HUE smart LED lamp, and Amazon Echo Home device. Furthermore, various benign and malicious network traffic were generated, including Okiru and Torii botnet attacks, as depicted in Figure 4 and attack descriptions in Table 3.

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

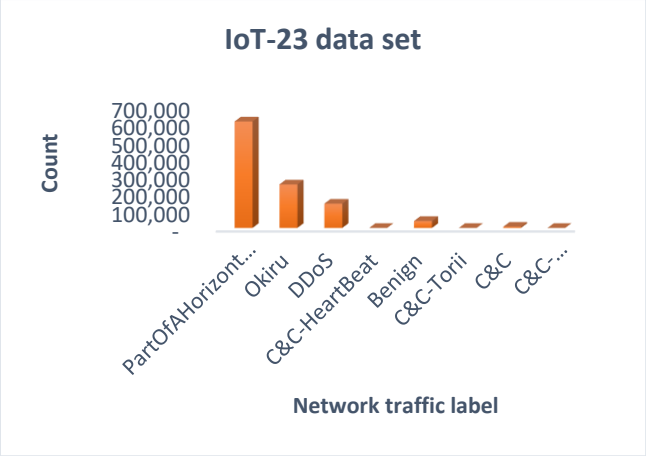

Figure 4. The IoT-23 data set sample

The description for these attacks is briefly explained in the below table.

Table 3: IoT-23 network traffic descriptions

Network traffic	**Description**
Benign	"Normal network traffic, i.e., packets."
PartOfHorizontalPortScan	"An attack launched by collecting information through horizontal port scan."
Okiru	"Okiru botnet attack."
DDoS	"An attack launched to deny multiple authorised devices access to a network."
C&C – HeartBeat	"An attack launched from a compromised device and kept "alive" by connecting it to a infected C&C server."
C&C - Torii	"Torii botnet attack."
C&C	"An attack launched from a C&C server."
C&C Filedownload	"Files downloaded to infected devices that had connections to a C&C server."

The IoTID-20 data set is presented next.

IoTID-20 data set

This data set consisted of two smart home devices: SKT NGU and EZVIZ Wi-Fi cameras. Other devices connected to the IoT networks include smartphones, tablets, and laptops. Furthermore, benign and malicious network traffic has been generated, including various types of Mirai botnet attacks.

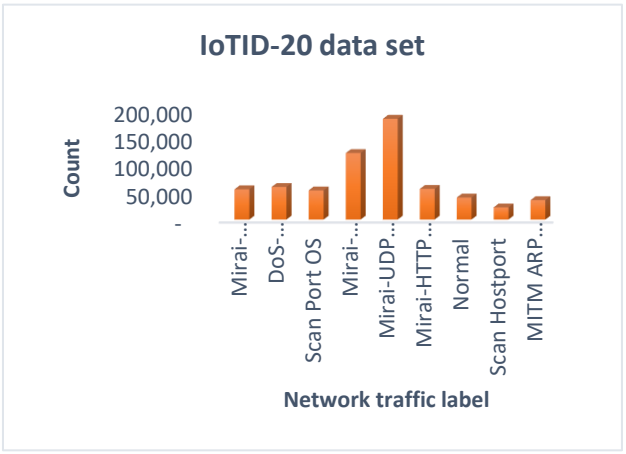

Figure 5. The IoTID-20 data set sample

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

Furthermore, the distribution of the number of attacks in

Figure **5** is imbalanced with the Mirai-UDP flooding attack in most instances. The descriptions of these attacks are briefly explained in Table 4.

Table 4: The IoTID-20 network traffic descriptions

Network traffic	**Description**
Benign	"Normal network traffic, i.e., packets."
Mirai-Ackflooding	"Mirai botnet attack using ack flooding attack."
DoS-Synflooding	"An attack that overwhelms a server connection requests such that legitimate users cannot access the server."
Scan Port OS	"An attack that scans open vulnerable ports in a network."
Mirai-Hostbruteforce	"Mirai botnet attack using host brute force attack."
Mirai-UDP flooding	"Mirai botnet attack using UDP flooding attack."
Mirai-HTTP flooding	"Mirai botnet attack using HTTP flooding attack."
Scan Hostport	"An attack that scans vulnerable host port to attack in a network."
MITM ARP spoofing	"A man in the middle (MITM) attack that intercepts communication between users and servers."

The TON_IoT-20 data set is presented next.

TON_IoT-20 data set

The TON_IoT-20 data set was developed at UNSW Canberra IoT Labs and Cyber Range. The authors used various devices, such as smart TVs and laptops, including Windows operating systems. The TON_IOT-20 data sets consist of benign and nine botnet attacks, including DDoS and MITM described by forty-five features. Figure 6 depicts the number of instances of network traffic labels.

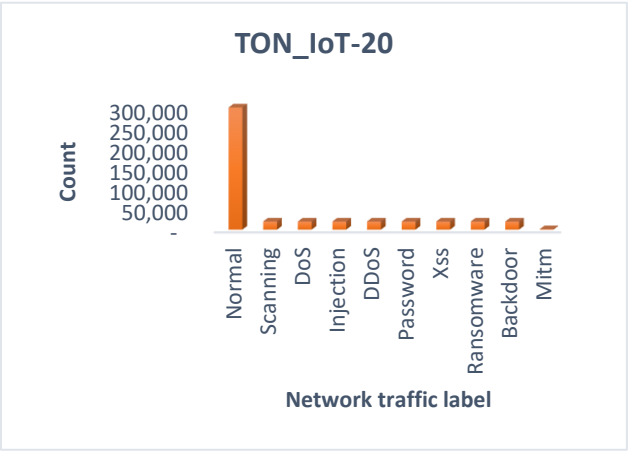

Figure 6. The TON_IoT-20 data set sample

The TON_IoT-20 data set was imbalanced, as shown in Figure 6 with normal network traffic being the majority class. The descriptions of these attacks are briefly explained in Table 5.

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

Table 5: The TON_IoT-20 network traffic descriptions

Network traffic	Description
Benign	"Normal network traffic, i.e., packets."
Scanning	"An attack that scans for open vulnerable ports in a network."
DoS	"An attack that denies legitimate users access to a network."
Injection	"An attack that injects fake/deceptive data from user's machine."
DDoS	"An attack launched to deny multiple authorised devices access to a network."
Password	"Any hacking attack such as brute-force attack to guess users password."
Cross-site scripting (Xss)	"An attack that injects malicious executable program into a code of a trusted source such as a software."
Ransomware	"Any malware attack whereby an attacker demands a ransom from a victim to de-crypt a malware."
Backdoor	"An attack that bypasses standard network security controls."
Mitm	"A man in the middle (MITM) attack that intercepts communication between users and servers."

The N-BaIoT data set is presented next.

N-BaIoT data set

The N-BaIoT data set was developed in 2018 using nine standard IoT devices such as security cameras and webcams, which are infected by various types of botnet attacks. This data set provides traces, that is, features of network traffic for each device. Mirai infected IoT devices and bashlite (Gafgyt) botnet attacks. Mirai and Bashlite are two common malware attacks in IoT networks, with the Mirai botnet attack first witnessed in 2016 launching DDoS attacks, whereas Bashlite was first witnessed in 2014, affecting Linux-based systems.

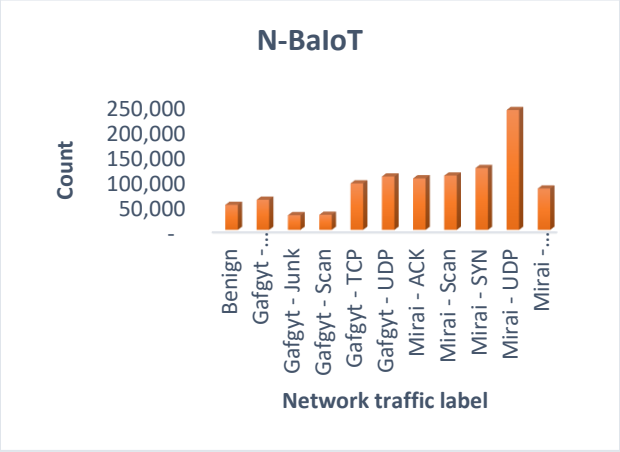

Figure 7. The N-BaIoT data set sample

The N-BaIoT data set was imbalanced, as shown in Figure 7 with MiraiUDP traffic being the majority class. The descriptions of these attacks are briefly explained in Table 6.

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

Table 6: The N-BaIoT network traffic descriptions

Network traffic	Description
Benign	"Normal network traffic, i.e., packets."
Gafgyt - Combo	"Gafgyt botnet attack launched by sending spam data and creating fake connections in a network."
Gafgyt - Junk	"Gafgyt botnet attack launched by flooding the network with fake/junk."
Gafgyt - Scan	"Gafgyt botnet attack launched by scanning the network for vulnerable ports."
Gafgyt - TCP	"Gafgyt botnet attack launched by flooding TCP protocol."
Gafgyt - UDP	"Gafgyt botnet attack launched by flooding UDP protocol."
Mirai - ACK	"Mirai botnet attack launched by flooding a server with ACK packets."
Mirai - Scan	"Mirai botnet attack launched by scanning a network for vulnerable ports."
Mirai - SYN	"Mirai botnet attack launched by flooding a server with SYN packets."
Mirai - UDP	"Mirai botnet attack launched by flooding a server with UDP protocol."
Mirai - UDPPlain	"Mirai botnet attack launched by flooding a server with plain UDP protocol."

Having introduced the data sets used in this study, we implemented the CySecML methodology to intelligently discover unknown botnet attacks. The data storage and quality check blocks are shown in Figure 3. Data storage and quality checks were conducted.

4.1 Data storage and quality checks experiments

The IoT-23, IoTID-20, TON_IoT-20, and N-BaIoT data sets were downloaded from the internet and saved locally. As illustrated in Figure 4 and Figure 7, the sample sizes for these data sets were sufficiently large; therefore, no additional samples were required. Data quality characteristics for cybersecurity data sets including availability, usability, reliability, relevance, data size, and presentation quality, were proposed in our previous work. This study proposes a simple data quality score (DQS) approach in which each data characteristic is assigned a value of 0, 0.5, and 1 – poor, adequate, and excellent, respectively. The scores were assigned based on the analysis of each data set. Scorecard approaches, such as the DQS, generally answer binary types of questions, that is, "yes or no, where these questions can be answered with very high (score of 1), medium (score of 0.5), and very low (score of 0) that are linearly aggregated. DQS is expressed as follows:

$$DQS = availability + usability + reliability + relevance + data\ size + presentation\ quality$$

Where DQS of ≥ 5 is considered quality, else poor quality.

Table 7: Data quality characteristics

Data set	Availability	Usability	Reliability	Relevance	Data size	Presentation quality	Total
IoT-23	1	0.5	1	1	1	1	5.5
IoTID-20	1	0.5	1	1	1	1	5.5
TON_IoT-20	1	0.5	1	1	1	1	5.5
N-BaIoT	1	0.5	1	1	1	0.5	5

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

Overall, as presented in Table 7, the data sets used in this study are of high quality. Therefore, experimental findings based on these parameters can be considered dependable. All data sets used in this study were rated with a score of 1 for availability characteristics, as they were easily accessible on the Internet. For usability, they all scored 0.5, because they did not contain unknown botnet attacks. In terms of reliability, relevance, and data size, they all scored 1 because they contained sufficient instances of various botnet attacks that could be used to train ML algorithms to effectively discover unknown botnet attacks. We scored the presentation quality of the N-BaIoT data set (0.5) because packet capture (pcap) files are stored as different CSV files. The presentation format of a pcap file is not ideal for feature selection. The data cleaning and feature selection blocks are implemented next.

4.2 Data cleaning and feature selection experiments

Data cleaning is the process of correcting corrupt data or data errors such as blank fields in a data set. Specifically, negative values in the features were removed because Benford's law (BL) is applicable to real positive numbers. A new unknown attack can be created manually by combining attacks across the same features. For example, consider the IoT-23 data set, an unknown attack = Okiru+DDoS + C&C-HeartBeat + C&C-Tori + C&C + C&C-Filedownload + PartOfAHorizontal PortScan. Simply put, BL uniquely identifies significant features by analysing significant leading digits of data using the first digit test (FDT), second digit test (SDT), third digit test (TDT), first two-digit test (F2DT), and last two-digit test (L2DT). If a data set violates the BL distribution tests, it is considered anomalous. The Null hypothesis (H_0) is that a distribution obeys the BL, and the alternative hypothesis (H_1) is that a distribution violates the BL. If the p-value was < 0.05, H_0 was rejected; otherwise, H_0 was not rejected. The results of the BL for all data sets used in this study are presented in Table 8. A feature is considered significant if it obeys one of the BL distributions on the benign data, while simultaneously violating all remaining BL distributions on the malicious data.

Table 8: Feature selection results using Benford's law

Data set	Significant features
IoT-23	"id.orig_p, duration, orig_bytes, resp_bytes, orig_pkts, resp_pkts, resp_ip_bytes"
IoTID-20	"Fwd_IAT_Mean, Fwd_IAT_Max, Fwd_IAT_Min, Bwd_IAT_Min, Fwd_Pkts/s"
TON_IoT-20	"duration, dst_bytes, missed_bytes, src_pkts, dst_pkts, dst_ip_bytes, http_response_body_len"
N-BaIoT	"MI_dir_L0.1_variance, MI_dir_L0.01_mean, MI_dir_L0.01_variance, H_L5_variance, H_L0.1_mean, H_L0.1_variance, H_L0.01_mean, H_L0.01_variance, HH_L1_radius, HH_L0.01_radius, HH_jit_L0.01_mean, HH_jit_L0.01_variance"

For the IoT-23 data set, we identified seven significant features out of a total of 15 features. BL identified features such as duration (total flow of duration network traffic), orig_bytes (number of payload bytes sent as the originator), and orig_pkts (number of packets sent by the originator) as significant in differentiating between benign and botnet attacks. This data set contained a couple of categorical features, such as proto (transactional protocol) and history (state of connection history); thus, these were found to be redundant because they failed to differentiate between benign and botnet attacks. Our findings were consistent with those reported by Abdalgawad et al. [11] They identified eight significant features on the IoT-23 data set: proto (transaction protocol: icmp, tcp, udp), service (dhcp,http, dns, ssl, irc, ssh), orig_bytes, resp_bytes, and duration. In our opinion, features such as proto-and-service, which are assigned categorical values, cannot distinctively differentiate between benign and malicious network traffic.

In the IoTID-20 data set, we identified five significant features out of 82. The majority of the features in this data set were binary, that is, {0,1}, or did not span {0,1,..,9}; therefore, BL could not differentiate between benign and botnet attacks. The significant features identified include Fwd_IAT_mean (mean time between two packets sent in the forward direction), Bwd_IAT_Min (minimum time between two packets sent in the backward direction), and Fwd_Pkts/s (number of forward packets per second), which are consistent with those identified by Ahmad et al. [16] using the mutual information feature selection method. In the TON_IoT-20 data set, we identified seven significant features from 41 features, including duration, dst_bytes, and missed_bytes. Our results were consistent with those reported by Krishan et al. [8]. Similar to the IoTID-20 data set, the majority of the features in this data set were binary or did not span the {0,1,…,9} dimension. Interestingly, features related to the duration, packets (src_pkts, dst_pkts), and bytes (dst_bytes, missed_bytes, dst_ip_bytes) were significant, as in the IoT-23 data set.

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

Finally, for the N-BaIoT data set, we identified 12 significant features from the 115 features. Similarly, the majority of features in this data set did not span the {0,1,…,9} dimension; therefore, BL could not differentiate between benign and botnet attacks. A comparison of our results with those of Alazzam et al. [24] identified ten significant features. In this instance, the significant features we identified differed from those identified by Alazzam et al. [24] where they identified "weight" related features such as "MI_dir_L3_weight" and "H_L1_weight" to be significant, Figure 8 and Figure 9 illustrate BL FDT using the destination byte feature, which obeys BL on benign traffic, whereas it is violated on malicious (i.e., unknown) botnet traffic.

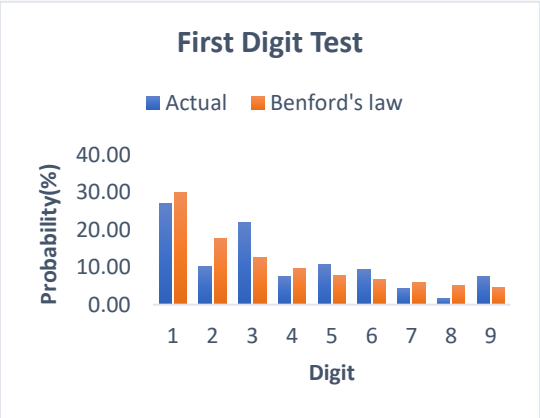

Figure 8. Destination bytes feature – benign traffic

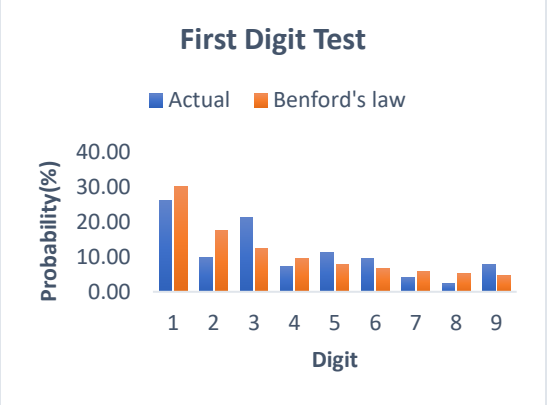

Figure 9. Destination bytes feature – malicious traffic

The significant features identified in Table 8 were used as inputs to the ML algorithms, as implemented in the next section.

4.3 Machine learning and decision-making experiments

This section discusses the implementation of machine learning (ML) algorithms to intelligently discover unknown botnet attacks. Four popular semi-supervised ML algorithms are used in this study: self-training, co-training, label spreading, and label propagation. Semi-supervised ML algorithms were chosen in this study because of their ability to work with labelled data that is, labelled attacks or benign network traffic and unlabelled data, that is, unknown botnet attacks used to fine-tune the ML algorithm. In summary, self-training ML algorithms uses a supervised ML algorithm (Gaussian Naïve Bayes in this study) on a small portion of labelled data and learns to predict labels of this data set; this becomes the "base case" of the model. Thereafter, the base-case model was used to label large portions of the unlabelled data set (i.e., pseudo-labels). This process was repeated until all data were labelled and the model selected data points labelled with high confidence. Finally, the original labelled and pseudo-labelled data points were used to improve the performance of the ML model. Co-training is an improved version of self-training because it uses two supervised ML classifiers based on "two views" of a data set. The views of a data set were based on different sets of features from the same data set. Label propagation is a graph-based semi-supervised ML that relies on clustering and smoothness assumptions such that data points that are

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

"close" to each other will have the same label, whereas data points further from this will have a "different" label. Label spreading is an improved version of label propagation given that labels may be updated during the iteration process. The performance of these algorithms was determined using the following measures: a 30%-70% (train-test) split was used in this study for each data set.

4.3.1 Machine learning evaluation measures

Machine learning evaluation measures used in this study are stated below.

Table 9: Confusion matrix

	Actual malicious	**Actual benign**
Predicted malicious	True positive (TP)	False positive (FP)
Predicted benign	False negative (FN)	True negative (TN)

From Table 9, the below evaluation measures can be derived.

Precision is the ratio of true positives to the proportion of the predicted positive results.

$$Precision = \frac{TP}{TP + FP}$$

The recall or true positive rate (TPR) is the ratio of true positives to the actual positive results.

$$Recall = \frac{TP}{TP + FN}$$

F_1-score represents the harmonic mean of precision and recall.

$$F_1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

MCC-score measures the correlation between predicted and actual cases.

$$MCC - score = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Table 10: Semi-supervised machine learning algorithms using IoT-23 data set

SSML Algorithm	**Precision (%)**	**Recall (%)**	**F_1-score (%)**	**MCC-score (%)**
Self-training	94	93	93	91
Co-training	82	74	73	71
Label spreading	67	97	72	70
Label propagation	67	97	72	70

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

Table 11: Semi-supervised machine learning algorithms using IoTID-20

SSML Algorithm	Precision (%)	Recall (%)	F_1-score (%)	MCC-score (%)
Self-training	94	98	94	92
Co-training	93	98	93	91
Label spreading	61	71	59	55
Label propagation	61	71	59	55

Table 12: Semi-supervised machine learning algorithms using TON_IoT-20

SSML Algorithm	Precision (%)	Recall (%)	F_1-score (%)	MCC-score (%)
Self-training	87	83	82	81
Co-training	76	86	74	71
Label spreading	73	87	76	72
Label propagation	73	87	76	72

Table 13: Semi-supervised machine learning algorithms using N-BaIoT

SSML Algorithm	Precision (%)	Recall (%)	F_1-score (%)	MCC-score (%)
Self-training	88	85	85	83
Co-training	78	88	76	72
Label spreading	71	83	72	71
Label propagation	71	83	72	71

4.4 Discussion of results

The previous section presented the results of the four semi-supervised machine learning (SSML) algorithms used in this study to intelligently discover unknown botnet attacks. Four data sets containing C&C botnet attacks are used in the experiments. For each data set, we evaluated the performance of the SSML algorithms based on the significant features identified by Benford's law (BL), and the results are presented in Table 10 to Table 13. Throughout our experiments, the self-training algorithm performed the best, with an MCC-score of 92% and an F_1-score of 94%. A co-training algorithm follows this. The label spreading (LS) and label propagation (LP) algorithms performed the least with the same results, suggesting that LS did not update its labels during the labelling iteration process.

In this setting, the self-training algorithm performed the best because of its ability to improve performance-using data labelled with high confidence. This process ensures that the testing stage includes only data points that the model can distinguish. Specifically, in the training phase of the self-training algorithm, unlabelled data (unknown botnet attack) traffic is labelled using a Gaussian Naïve Bayes supervised machine-learning algorithm, and only data points that are labelled with high confidence are included in the testing stage of the algorithm. The setup for the co-training algorithm was similar to that of the self-training algorithm; however, its performance was lower than that of its counterpart. One explanation for this is that co-training algorithms divide a data set into "two

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

views" using different significant features, whereas self-training algorithms use the same data set based on the identified significant features. Therefore, the differences between the data sets can influence the performances of the two SSML algorithms. Given that the data sets used in this study were of high quality, the results obtained can be relied upon.

The performance results achieved in this study are comparable with those of state-of-the-art studies discussed in the literature review section. For instance, Liu et al. [3] achieved an accuracy score of approximately 96% using deep learning (DL) algorithms supported by the random forest feature selection method. Abdalgawad et al. [11] used DL algorithms on the IoT-23 data set to achieve an F_1-score of 85%. Krishnan et al. [8] used DL algorithms to achieve an F_1-score of 93% on the TON_IoT-20 data set using a localized generalized matrix learning vector quantization feature-selection method. However, this study demonstrated that machine-learning algorithms, particularly self-training, are effective for the intelligent discovery of botnet attacks based on significant features to identify and trace botnet attacks. Finally, we examined the extent to which our research questions and objectives were addressed.

Using the Wilcoxon signed-rank test, the above SSML algorithms were evaluated in terms of F_1-scores to determine the difference in the performance of the two algorithms. In this case, a low p-value score suggests a difference in the algorithm performance, whereas a high p-value score suggests a strong similarity in performance. It is evident in

Figure **10** that the self-training SSML algorithm outperforms all the algorithms, whereas label spreading and label propagation achieve similar results.

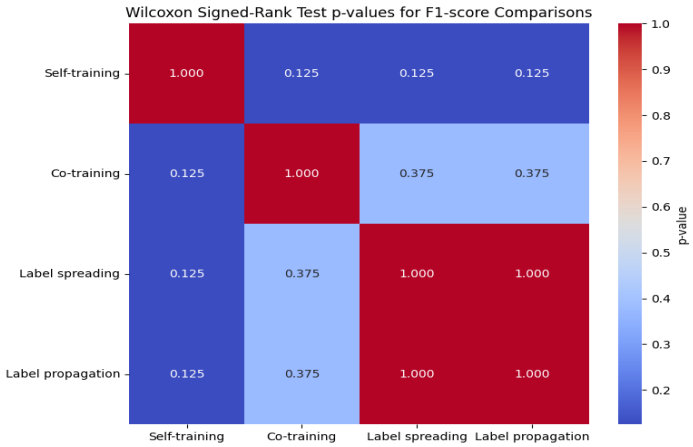

Figure 10. The Wilcoxon signed-rank test results

Research question 1:

What are the prominent cyber threats posed by botnets in IoT?

The literature review presented in Section 2 highlights various botnet attacks, including DDoS, scanning, brute force, spoofing, man-in-the-middle, and unknown attacks. Botmasters can launch these attacks in various ways, such as by using centralised and decentralised architectures. These attacks enable the botmaster to spread and compromise between multiple devices in a network.

Research question 2:

What are the current trends and challenges in discovering botnet attacks?

Centralised and decentralised are two main types of botnet architectures, whereby machine learning approaches such as deep learning, semi-supervised, and graph-based algorithms are commonly used to discover centralised botnets, whereas federated learning approaches are used to discover decentralised botnets. The main challenge facing centralised botnet architectures is that data are stored on a single server, which can cause ML algorithms built upon this system to be heavyweight, owing to voluminous data. Therefore, most studies based on this approach have aimed to propose lightweight ML based cybersecurity solutions. In federated learning approaches, data are stored on multiple servers; this naturally reduces the burden of dealing with voluminous data and improves data privacy and security in instances of data leakage. However, the main challenge with federated learning is that data across multiple servers may not be the same; therefore, different algorithms may be required to effectively discover botnet attacks. The ML/DL algorithms used to discover unknown botnet attacks were also highlighted by Ahmad et al. [25] to discover unknown bot attacks.

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

Research question 3:

Which features are significant for the tracing and intelligent discovery of unknown botnet attacks?

In this study, state-of-the-art data sets containing various types of botnet attacks were used. Known botnet attacks were used to create unknown botnet attacks. Benford's law (BL) was used in this study to identify significant features for tracing botnet attacks, including duration, packet information (e.g., orig_pkts and resp_pkts), and byte-related information (e.g., dst_bytes and missed_bytes), as shown in Table 8.

Research question 4:

How can machine-learning-based cybersecurity solutions be enhanced to effectively discover unknown centralised botnet attacks?

A common approach for enhancing ML based cybersecurity solutions is to reduce the computational cost of machine learning-based cybersecurity solutions. Feature selection is crucial in lightweight ML based cybersecurity solutions. Feature selection can reduce the training time of an ML algorithm, reduce overfitting, and improve overall performance. In this study, Benford's law (BL) was used, which is based on straightforward mathematical equations, unlike PCA and localized generalized matrix learning vector quantization feature selection methods that are generally computationally expensive. Therefore, feature selection methods such as BL can make cybersecurity solutions lightweight in IoT environments where computational memory is often a challenge. In addition, BL is scalable in IoT environments because its computational cost does not increase as the data size increases.

In the next section, we conclude the paper and discuss possible future research directions.

5. Conclusion

A lightweight data-driven methodology called CySecML was implemented to intelligently discover centralised unknown botnet attacks in the IoT domain. This methodology is lightweight owing to its adoption of a computationally efficient feature-selection method based on Benford's law. Benford's law was applied to four different state-of-the-art data sets that consisted of various botnet attacks, including DDoS, Mirai, scanning, man in the middle, and scanning, among others. Unknown botnet attacks were created by combining known attacks in each data set. Benford's law found features related to packets, duration, and network bytes to be significant for differentiating between benign and unknown botnet attacks. Our findings regarding these significant features are consistent with those of the current study. Thereafter, the identified significant features were used as inputs into semi-supervised machine learning (SSML) algorithm to intelligently discover botnet attacks. Self-training SSML consistently performed well in the experiments, achieving the best results with an F_1-score of 94%.

Although our study demonstrated promising results for discovering unknown botnet attacks, the experiments were not conducted in a live IoT environment but in data extracted from IoT environments. Our future work will include applying the CySecML methodology to a live environment by adopting reinforcement learning, which will enable the system to learn on its own to intelligently discover unknown botnet attacks.

Funding: This research received no external funding.

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

References

[1] W. Fei, H. Ohno, and S. Sampalli, "A systematic review of IoT security: Research potential, challenges, and future directions," *ACM Comput. Surv.*, vol. 56, no. 4, pp. 1–40, 2023, doi: 10.1145/3625094.

[2] R. Saadouni, C. Gherbi, Z. Aliouat, Y. Harbi, and A. Khacha, "Intrusion detection systems for IoT based on bio-inspired and machine learning techniques: A systematic review of the literature," *Cluster Comput.*, 2024, doi: 10.1007/s10586-024-04388-5.

[3] Z. Liu, N. Thapa, A. Shaver, K. Roy, M. Siddula, X. Yuan, and A. Yu, "Using embedded feature selection and CNN for classification on CCD-INID-V1—A new IoT data set," *Sensors*, vol. 21, no. 14, p. 4834, 2021, doi: 10.3390/s21144834.

[4] J. Zhang, S. Liang, F. Ye, R. Q. Hu, and Y. Qian, "Towards detection of zero-day botnet attack in IoT networks using federated learning," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2023, pp. 763–768.

[5] T. Ohtani, R. Yamamoto, and S. Ohzahata, "IDAC: Federated learning-based intrusion detection using autonomously extracted anomalies in IoT," *Sensors*, vol. 24, no. 10, p. 3218, 2024, doi: 10.3390/s24103218.

[6] H. Alkahtani and T. H. H. Aldhyani, "Botnet attack detection by using CNN-LSTM model for Internet of Things applications," *Secur. Commun. Netw*, vol. 2021, p. 3806459, 2021, doi: 10.1155/2021/3806459.

[7] V. Kumar and D. Sinha, "A robust intelligent zero-day cyber-attack detection technique," *Complex Intell. Syst.*, vol. 7, no. 5, pp. 2211–2234, 2021, doi: 10.1007/s40747-021-00396-9.

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025

[8] D. Krishnan and P. Shrinath, "Robust botnet detection approach for known and unknown attacks in IoT networks using stacked multi-classifier and adaptive thresholding," *Arab. J. Sci. Eng.*, 2024, doi: 10.1007/s13369-024-08742-y.

[9] A. H. Celdrán, P. M. S. Sánchez, M. A. Castillo, G. Bovet, G. M. Pérez, and B. Stiller, "Intelligent and behavioral-based detection of malware in IoT spectrum sensors," *Int. J. Inf. Secur.*, vol. 22, no. 3, pp. 541–561, 2022, doi: 10.1007/s10207-022-00602-w.

[10] B. Bezawada, M. Bachani, J. Peterson, H. Shirazi, I. Ray, and I. Ray, "Behavioral fingerprinting of IoT devices," in *Proc. Workshop Attacks Solutions Hardware Secur. (ASHES)*, 2018, pp. 1–11.

[11] N. Abdalgawad, A. Sajun, Y. Kaddoura, I. A. Zualkernan, and F. Aloul, "Generative deep learning to detect cyberattacks for the IoT-23 data set," *IEEE Access*, vol. 10, pp. 6430–6441, 2022, doi: 10.1109/ACCESS.2021.3140015.

[12] S. I. Popoola, R. Ande, B. Adebisi, G. Gui, M. Hammoudeh, and O. Jogunola, "Federated deep learning for zero-day botnet attack detection in IoT-edge devices," *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3930–3944, 2022, doi: 10.1109/JIOT.2021.3100755.

[13] B. I. Hairab, H. K. Aslan, M. S. Elsayed, A. D. Jurcut, and M. A. Azer, "Anomaly detection of zero-day attacks based on CNN and regularization techniques," *Electronics*, vol. 12, no. 3, p. 573, 2023, doi: 10.3390/electronics12030573.

[14] A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, and A. Alazab, "A novel ensemble of hybrid intrusion detection system for detecting Internet of Things attacks," *Electronics*, vol. 8, no. 11, p. 1210, 2019, doi: 10.3390/electronics8111210.

[15] A. Alharbi and K. Alsubhi, "Botnet detection approach using graph-based machine learning," *IEEE Access*, vol. 9, pp. 99166–99180, 2021, doi: 10.1109/ACCESS.2021.3094183.

[16] Z. Ahmad, A. S. Khan, K. Nisar, I. Haider, R. Hassan, M. R. Haque, S. Tarmizi, and J. J. P. C. Rodrigues, "Anomaly detection using deep neural network for IoT architecture," *Appl. Sci.*, vol. 11, no. 15, p. 7050, 2021, doi: 10.3390/app11157050.

[17] K. Roshan and A. Zafar, "An optimized auto-encoder based approach for detecting zero-day cyber-attacks in computer network," in *Proc. 5th Int. Conf. Inf. Syst. Comput. Netw. (ISCON)*, 2021, pp. 1–6.

[18] R. Li, Q. Li, J. Zhou, and Y. Jiang, "ADRIoT: An edge-assisted anomaly detection framework against IoT-based network attacks," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 10576–10587, 2022, doi: 10.1109/JIOT.2021.3122148.

[19] D. Xu and Y. Tian, "A comprehensive survey of clustering algorithms," *Ann. Data Sci.*, vol. 2, no. 2, pp. 165–193, 2015, doi: 10.1007/s40745-015-0040-1.

[20] X. Xu, H. Liu, and M. Yao, "Recent progress of anomaly detection," *Complexity*, vol. 2019, p. 2686378, 2019, doi: 10.1155/2019/2686378.

[21] C. D. Luu, V. Q. Nguyen, T. S. Pham, and N.-A. Le-Khac, "A zero-shot deep learning approach for unknown IoT botnet attack detection," in *Proc. RIVF Int. Conf. Comput. Commun. Technol. (RIVF)*, 2023, pp. 474–479.

[22] A. Yulianto, P. Sukarno, and N. A. Suwastika, "Improving AdaBoost-based intrusion detection system (IDS) performance on CIC IDS 2017 data set," *J. Phys.: Conf. Ser.*, vol. 1192, p. 012018, 2019, doi: 10.1088/1742-6596/1192/1/012018.

[23] X. Liang and T. Znati, "A long short-term memory enabled framework for DDoS detection," in *Proc. IEEE Glob. Commun. Conf. (GLOBECOM)*, 2019, pp. 1–6.

[24] H. Alazzam, A. Alsmady, and A. Al Shorman, "Supervised detection of IoT botnet attacks," in Proc. 2nd Int. Conf. Data Sci., E-Learning Inf. Syst., 2019, pp. 1–5.

[25] R. Ahmad, I. Alsmadi, W. Alhamdani, and L. Tawalbeh, "Zero-day attack detection: a systematic literature review," *Artif. Intell. Rev.*, vol. 56, no. 9, pp. 10733–10811, 2023, doi: 10.1007/s10462-023-10437-z.

[26] A. Blaise, M. Bouet, V. Conan, and S. Secci, "Detection of zero-day attacks: An unsupervised port-based approach," *Comput. Netw*, vol. 180, p. 107391, 2020, doi: 10.1016/j.comnet.2020.107391.

[27] S. Li, Y. Cao, S. Liu, Y. Lai, Y. Zhu, and N. Ahmad, "HDA-IDS: A hybrid DoS attacks intrusion detection system for IoT by using semi-supervised CL-GAN," *Expert Syst. Appl.*, vol. 238, p. 122198, 2024, doi: 10.1016/j.eswa.2023.122198.

[28] N. A, H. J, S. P. S. Prakash, and K. Krinkin, "Class imbalance and concept drift invariant online botnet threat detection framework for heterogeneous IoT edge," *Comput. Secur.*, vol. 141, p. 103820, 2024, doi: 10.1016/j.cose.2024.103820.

[29] J. Wang, S. Lu, S.-H. Wang, and Y.-D. Zhang, "A review on extreme learning machine," *Multimed. Tools Appl.*, vol. 81, no. 28, pp. 41611–41660, 2022, doi: 10.1007/s11042-021-11007-7.

[30] S. Khan and A. B. Mailewa, "Discover botnets in IoT sensor networks: A lightweight deep learning framework with hybrid self-organizing maps," *Microprocess. Microsyst*, vol. 97, p. 104753, 2023, doi: 10.1016/j.micpro.2022.104753.

DOI: https://doi.org/10.54216/JCIM.170209
Received: February 25, 2025 Revised: June 05, 2025 Accepted: July 27, 2025