A Review on Developing a System for Intelligent Intrusion Classification for Internet of Things Gateway Communication

***AAR Senthil Kumaar, **Dr. U. Moulali**
*Research Scholar, **Supervisor
Department of Computer Science
Faculty of Computing & Information Technology
Himalayan University, Itanagar, Arunachal Pradesh, India

[1]*Received: 29 January 2025; Accepted: 07 April 2025; Published: 28 April 2025*

ABSTRACT

Internet of Things (IoT) gateways are both crucial control points and potential weak spots due to the extremely diverse and dispersed communication landscape brought forth by the exponential expansion of IoT deployments. Traditional rule-based security measures are frequently insufficient to guarantee strong defence against intricate intrusion patterns due to the ever-increasing size and complexity of cyber threats. In order to protect communication across IoT gateways, this research details a comprehensive solution for intelligent intrusion classification that makes use of machine learning methods. The suggested design incorporates adaptive machine learning classifiers, multi-stage feature engineering, and continuous network traffic acquisition to identify unusual behaviours more accurately. Normal and malicious traffic patterns are characterized by extracting a complete collection of features, which includes attributes pertaining to time, statistics, protocols, and flows. The appropriateness of different supervised models for real-time intrusion detection on IoT devices with low resources is assessed. These models include Random Forest, Gradient Boosting, Support Vector Machine, and classifiers based on deep learning. In addition, the system optimizes hyper parameters and uses feature selection to maximize classification performance with little computing overhead. The system that is powered by machine learning is far better at detecting attacks like denial of service, spoofing, data injection, replay, and unauthorized access attempts than traditional detection approaches. This was confirmed experimentally using typical Internet of Things intrusion datasets and simulated gateway traffic. The accuracy, F1-score, detection delay, and false-alarm rate are performance measures that show how well the system works in limited IoT settings. In sum, the suggested intelligent intrusion categorization framework provides a resilient, scalable, and energy-efficient means of strengthening the safety and reliability of communications between IoT gateways. By strategically combining machine learning with lightweight analytics, it makes a significant contribution to the development of next-generation IoT security infrastructures. In order to create intrusion classification systems that are optimal for resource-constrained IoT gateways, this review points out the shortcomings of current methods and lays out the paths for future research.

Keywords: *Internet of Things; Accuracy; F1-Score; Random Forest; Gradient Boosting; Support Vector Machine*

1. Introduction

The expansion of Internet of Things (IoT) technologies has revolutionized sectors including healthcare, agriculture, smart homes, and industrial automation. IoT gateways function as essential elements in this architecture, facilitating efficient data transmission, device administration, and protocol conversion. Nonetheless, their posture renders them principal targets for assailants. Intrusions including Denial of Service (DoS), spoofing, data manipulation, replay assaults, and virus dissemination present significant threats to the integrity and security of IoT applications. Conventional security measures, such as firewalls and signature-based intrusion detection systems, frequently

[1] *How to cite the article:* Kumaar A.A.R.S, Moulali U (2025); A Review on Developing a System for Intelligent Intrusion Classification for Internet of Things Gateway Communication; *International Journal of Technology, Science and Engineering*; Vol 8 Issue 2; 8-17

inadequately identify emerging and unfamiliar threats. Machine learning (ML) has emerged as a viable solution, capable of discerning behavioral patterns and adeptly classifying hostile behaviors in real time. This research examines the advanced machine learning-based methods for intrusion classification specifically designed for IoT gateway communication. IoT systems face numerous security challenges due to their distributed architecture, resource limitations, and heterogeneity. Key challenges include:

- Limited computational resources restricting complex security algorithms.
- Gateway-level vulnerabilities enabling large-scale compromise.
- Enormous traffic volume and variability.
- Lack of standardized security protocols across device manufacturers.
- Physical accessibility leading to tampering risks.
- High dependency on wireless channels prone to interception.

These challenges necessitate intelligent, lightweight, and adaptive intrusion classification models. Machine learning has proven effective for intrusion classification due to its ability to generalize patterns and detect anomalies in network traffic. Commonly used ML approaches include:

- **Supervised Learning:** Algorithms such as Random Forest, Support Vector Machines, Gradient Boosting, and Neural Networks are widely used for attack classification.
- **Unsupervised Learning:** Clustering methods like k-means and DBSCAN help detect unknown intrusion patterns without labeled datasets.
- **Deep Learning:** Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), LSTM, and auto encoders provide high accuracy in complex traffic pattern recognition.
- **Hybrid Learning:** Combining ML and DL models yields enhanced accuracy and reduced false alarms.

Each approach comes with trade-offs related to model complexity, training cost, and suitability for resource-constrained IoT gateways. Most machine-learning-based intrusion classification systems for IoT gateways share a common pipeline:

- **Data Collection:** Acquiring real-time or historical network traffic at gateway nodes.
- **Preprocessing:** Cleaning, normalization, and handling missing values.
- **Feature Extraction:** Generating statistical, protocol-level, and temporal features.
- **Feature Selection:** Eliminating redundant features to reduce computation.
- **Model Training:** Building ML/DL classifiers and optimizing hyper parameters.
- **Deployment:** Integrating the trained model into lightweight gateway hardware.
- **Real-time Detection:** Monitoring live communication and classifying intrusions.

Different research works propose modifications to enhance efficiency, accuracy, and scalability.

2. **Neural Network Architecture**

- **Various artificial neural network (ANN)**

Various architectures have been introduced in the literature, and this section will provide a brief overview of some of them, with a particular focus on those extensively utilized in this thesis.

- **Convolutional Neural Networks (CNN)**

Convolutional Neural Networks (CNNs) are a specialized type of Feedforward Neural Networks (FNNs) designed to process grid-like data, such as images (2D grids) or time series (1D grids). Unlike traditional neural networks, CNNs utilize convolution operations instead of general matrix multiplication in at least one of their layers. Their architecture, characterized by shared weights and translation invariance, makes them highly effective for tasks such as image classification, object detection, and image segmentation in computer vision.

- **Recurrent Neural Networks (RNN)**

RNNs are tailored for sequential data like time series or texts. In an RNN, each time step receives input corresponding to that step and its output from the previous step. This recurrent nature provides the network with memory, as the output at a given time step depends on all previous inputs. Variants of RNNs that improve long-term memory include Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) networks.

- **Auto-encoders**

Auto-encoders are unsupervised neural networks that aim to copy their inputs to their outputs while adhering to certain constraints. They consist of an encoder, which compresses the input into a latent representation, and a decoder, which is responsible for reconstructing the original input. Auto encoders are trained to minimize reconstruction errors, making them useful for anomaly detection tasks.

- **Vibrational Auto-encoders (VAE)**

VAEs are a special type of auto-encoder. This introduces stochastic, as the same input can result in different latent vectors. VAEs are often used as generative models and contain a Kullback-Leibler (KL) divergence term in their loss function to force the latent space to follow a standard normal distribution.

- **Generative Adversarial Networks (GAN)**

GANs consist of two competing neural networks: discriminators and generators. The generator's role is to produce observations that closely resemble instances in the training set to deceive the discriminator, while the discriminator aims to distinguish real from generated cases. GANs are trained by iteratively improving the generator and discriminator until the generator creates indistinguishable instances from real ones. These ANN architectures serve various purposes and have been applied to different machine learning tasks, making them essential tools in deep learning and artificial intelligence.

- **Cyber-security Applications of Deep Learning**

Machine learning is applicable across various sub-fields of cybersecurity. Here are some notable examples:

3. **Malware Classification**

B. Kolosnjaji et al. [57] employed call sequences for the malware classification system. They conducted dynamic malware analysis, capturing the sequences of system calls made by malware in a controlled environment. These sequences were represented using one-hot encoded vectors. Their neural network architecture comprised a stacked

recurrent layer on top of a conventional layer, achieving an 89.4% and 85.6% recall of classification precision.

R. Pascanu et al. [59] introduced the use of Echo State Networks (ESN), a type of RNN, to detect malware based on sequences of system calls. Their RNN was trained unsupervised, followed by a feedforward neural network (FNN) for classification. Their results showed a True Positive Rate (TPR) of 98.3% and a False Positive Rate (FPR) of 0.1%.

Z. Li et al. [61] proposed a deep learning-based framework for detecting vulnerabilities in C/C++ programs at the source code level. They collected vulnerable code fragments and their patched versions from various vulnerability databases. They successfully identified vulnerabilities by tokenizing the code and employing a bidirectional LSTM classifier.

F. Yamaguchi et al. [62] proposed a vulnerability extrapolation method leveraging dominant API usage patterns. Their approach involved parsing source code into functions, extracting API symbols, and utilizing Principal Component Analysis (PCA) to examine the vector space. This technique enabled the identification of tasks with similar API usage, potentially revealing undiscovered vulnerabilities.

4. Network Intrusion Detection

Datasets like KDD99 and NSL-KDD, which describe TCP/IP communications, have been widely used in network intrusion detection. These datasets include duration, total bytes received and sent, and more advanced content-level features. They encompass two main classes: regular and attack network traffic, with attacks further categorized into DoS, R2L, U2R, and probes.

These examples highlight the versatility and effectiveness of deep learning models in addressing various cybersecurity challenges, from malware detection to vulnerability discovery and network intrusion detection.

5. Evolution of Intrusion Detection Systems (IDSs)

Intrusion Detection Systems (IDSs) have played a crucial role in network security by monitoring, analyzing, and detecting security threats through passive traffic collection and analysis. These systems aim to protect information systems' confidentiality, integrity, and availability by identifying potential intrusions [40, 41, 42]. The IDS framework typically operates in three stages: (1) the monitoring stage, which collects data using host-based or network-based sensors; (2) the analysis stage, which extracts relevant features and applies pattern recognition techniques; and (3) the detection stage, which utilizes either anomaly-based or misuse-based detection mechanisms [43].

The concept of IDS has evolved significantly over the last three decades. Denning (1987) introduced an intrusion detection model that compared system behavior against predefined standard patterns to detect malicious activity [44]. Axelsson (2000) surveyed 20 IDS research projects, categorizing them into host-based, network-based, and hybrid methods [45]. However, early IDS models relied heavily on local machine analysis rather than network traffic, limiting their effectiveness.

Subsequent research introduced intelligent techniques to enhance IDS capabilities. In 2013, M. Welling et al. [46] explored advanced feature selection and classification methods for intrusion detection, incorporating fuzzy logic, neural networks, genetic algorithms, and particle swarm intelligence to improve security and quality of service (quality of service). Their study analyzed 19 flow-based features, covering fundamental, packet content, and traffic attributes.

Mitchell and Chen (2014) reviewed 60 IDS research papers on wireless environments, including WLANs, WMNs, WPANs, WSNs, CPSs, ad hoc networks, and mobile telephony [47]. Their findings suggested that anomaly-based IDSs were well-suited for mobile telephony but suffered from high false favorable rates and computational complexity, leading to quality of service issues such as billing errors and packet delays. They also highlighted privacy concerns associated with packet-based analysis methods and emphasized detection latency as a key metric for future research.

That same year, E. Hodo et al. [48] examined IDS implementations in mobile ad hoc networks (MANETs) and wireless sensor networks (WSNs). They categorized security threats into passive and active attacks and emphasized the importance of IDSs in ensuring IOT security. Given the resource constraints of IOTs, they recommended low- power IDS solutions, proposing a hierarchical IDS model to optimize energy consumption. They also suggested different IDS architectures based on application needs: distributed IDS for mobile applications, centralized IDS for stationary networks, and hierarchical IDS for cluster-based networks.

Overall, the evolution of IDS research demonstrates ongoing advancements in detection methodologies, feature selection, and system optimization to address security challenges across diverse network environments.

6. Types and Methods of Intrusion Detection Systems (IDSs)

Implementing an Intrusion Detection System (IDS) varies based on the environment in which it is deployed. Broadly, IDSs can be categorized into host-based IDS (HIDS) and network-based IDS (NIDS), each serving distinct security functions. A Host-Based Intrusion Detection System (HIDS) is installed on a single system to monitor and protect it from malicious attacks that could compromise its operating system or data [49]. It primarily relies on host environment metrics, such as log files, system activity records, and user behavior patterns, which serve as input features for its decision engine [50]. Effective feature extraction from the host environment is fundamental to the operation of HIDS. The structural placement of a HIDS within a network ensures its ability to analyze internal system logs and detect potential intrusions before they escalate.

In contrast, a Network-Based Intrusion Detection System (NIDS) monitors network traffic by analyzing packets to detect anomalies and malicious activity [50]. NIDS can be either software-based or hardware-based. Snort is a widely used software- based NIDS, an open-source tool designed to analyze network traffic and detect security threats [51]. With the expansion of networks and increasing data traffic, hardware- based IDS solutions have become essential to ensure efficient real-time monitoring. Advanced hardware-based NIDS architectures, such as Field-Programmable Gate Arrays (FPGAs), offer significant advantages, including high-speed data processing, dynamic reconfiguration capabilities, and handling large volumes of network traffic [52, 53]. These characteristics make FPGAs a suitable choice for high-performance NIDS implementations.

7. Intrusion Detection Techniques

IDSs employ various detection techniques, relying on specialized algorithms to identify potential security threats. Several algorithms can be adapted to different detection methods, with many optimized for resource-efficient intrusion detection in IoT-based environments.

8. Lightweight Anomaly-Based Detection Algorithms

Anomaly-based IDS techniques focus on detecting deviations from normal behavior rather than relying on known attack signatures. A commonly used approach for anomaly detection is Principal Component Analysis (PCA), which serves as a dimensionality reduction and intrusion detection technique. PCA extracts a set of new variables by analyzing the variance-covariance structure of the original dataset. These new variables, or principal components, are linear combinations of the original variables and represent the most significant variations within the data [54, 55].

In IDS applications, PCA has been widely used for statistical modeling, data mining, and machine learning-based intrusion detection techniques. A. Jived et al. [56] employed PCA to develop an anomaly-based IDS that classifies network traffic into major and minor principal components. This system relies on the principal component scores to detect abnormal network behavior effectively. Other researchers have applied PCA in payload modeling, statistical intrusion detection, and machine learning-based threat analysis.

9. Misuse-Based Intrusion Detection

Misuse-based IDS, or signature-based IDS, relies on a predefined database of known attack signatures and malicious

code patterns to detect intrusions [59]. While effective in identifying previously documented threats, these systems face several limitations:

- Network Packet Overload – The extensive comparison of incoming packets against large signature databases can lead to processing delays and congestion.
- High Cost of Signature Matching – Continuous updates and signature- matching operations require significant computational resources.
- High False Alarm Rates – The reliance on signature-based detection often leads to false positives, especially when legitimate traffic resembles attack patterns.

Furthermore, specific network environments, such as Wireless Sensor Networks (WSNs), impose severe memory constraints, making traditional misuse-based IDS unsuitable due to their large storage requirements for attack signatures [61]. Additionally, these systems require constant updates to keep pace with evolving threats, highlighting their dependence on prior knowledge rather than real-time behavioral analysis.

Challenges and Constraints in Previous Research on Intrusion Detection within IoT Networks

Reference	Anomaly detection	No need to know the type of the device that is generating the traffic / Possibility to place the IDS outside the local network	Non- intrusive
R. Doshi et al. [107]	No	Yes	Yes
A. A. Diro et al. [108]	No	Yes	Yes
K. Yang et al. [142]	No	Yes	NA
E. Hodo et al. [113]	No	NA	NA
H. H. Pajouh et al. [110]	No	Yes	Yes
N. Mustafa et al. [133]	No	No	Yes
P. Shukla et al. [134]	No	NA	NA
Y. Meidan et al. [114]	Yes	No	Yes
Y. Mirsky et al. [115]	Yes	No	Yes
T. D. Nguyen et al. [116]	Yes	No	Yes
T. Luo et al. [117]	Yes	Yes	No

10. IoT Device Recognition:

The rapid expansion of IoT networks has led to several studies on IoT device fingerprinting. Through network traffic

analysis, researchers have explored machine learning-based approaches to classify and secure IoT devices.

Y. Meidan et al. proposed a machine learning-based network traffic analysis approach to identify IoT devices and create whitelists of authorized devices. Their method extracts features from full TCP sessions, ranging from SYN to FIN packets. Similarly, T. D. Nguyen et al. developed a system for detecting compromised IoT devices by leveraging the temporal periodicity of traffic generated by IoT devices. Their approach identifies devices based on periodic flow characteristics, accuracy, duration, and stability, using recurrent neural networks to detect deviations from normal behavior.

M. Miettinen et al. introduced a technique to classify IoT devices upon connection to a network, aiming to restrict the communication of vulnerable devices. Unlike other approaches, their method involves training a separate classifier for each device type using features extracted during the device setup phase. B. Bezwada et al. extended this concept by performing device behavioral fingerprinting, incorporating a subset of features from Miettinen et al. and payload-based features for classifier training. Additionally, R. Doshi et al. focused on IoT network security by performing network traffic classification to detect Dados attacks within IoT environments.

Apart from device fingerprinting, several studies have examined privacy concerns associated with IoT devices. A. Acer et al. demonstrated how an adversary could infer user activities within a smart home by profiling network traffic using machine learning algorithms. Similarly, N. Apthorpe et al. showed that analyzing network transmission and reception rates could expose sensitive user interactions with IoT devices. For example, monitoring the network activity of a Nest indoor security camera could reveal movement inside a home. Likewise, Copos et al. analyzed the network activity of smart devices such as the Nest Thermostat and Nest Protect, highlighting how an attacker could determine home occupancy based solely on network traffic patterns.

The proliferation of Internet of Things (IoT) devices has expanded the attack surface for cyber threats, necessitating advanced intrusion detection mechanisms. Traditional security measures often fall short in addressing the unique challenges posed by IoT networks, such as resource constraints, heterogeneity, and scalability. Researchers have increasingly turned to hybrid machine learning algorithms complemented by explainable artificial intelligence (XAI) techniques to enhance intrusion detection and classification within IoT ecosystems.

11. Hybrid Machine Learning Approaches in IoT Intrusion Detection

Hybrid Chinese learning approaches combine multiple algorithms to leverage their strengths, improving detection accuracy and robustness. For instance, Alqahtani et al.

[127] proposed a hybrid model integrating machine learning and deep learning techniques to enhance intrusion detection capabilities. This approach addresses the diverse and heterogeneous data transmitted across modern communication infrastructures, including IoT networks.

Similarly, a study by Alqahtani et al. [128] introduced a Hybrid Adaptive Ensemble for Intrusion Detection (HAEnID). This innovative method combines a series of multi- layered ensembles, including a Stacking Ensemble, Bayesian Model Averaging, and a Conditional Ensemble method. This hybrid approach aims to enhance intrusion detection while reducing false alarm rates.

In another study, a Hybrid Deep Learning Network Intrusion Detection System was proposed, employing a combination of Convolutional Neural Networks (CNNs) and Bidirectional Long long-term memory (BiLST networks to automatically extract spatial features and capture temporal dependency in network traffic data. The hybrid model demonstrated improved performance in detecting various intrusions within IoT networks.

12. Enhancing Interpretability with Explainable AI

While hybrid models enhance detection performance, their complexity often leads to a "black box" problem, where the decision-making process is not transparent. To address this, integrating XAI techniques has become crucial. Islam et al. [129] explored the infusion of domain knowledge into AI models to improve explainability. The researchers validated their approach on a network intrusion detection test case, demonstrating that incorporating domain knowledge provides better explainability and facilitates faster decision-making.

Another notable work by Dias et al. [130] proposed combining expert-written rules with dynamic knowledge generated by decision tree algorithms. This method aims to create an interpretable and explainable hybrid intrusion detection system, enhancing security by making the AI's decision-making process more transparent.

- Designing secure, low-energy data fusion algorithms for Internet of Things (IoT) networks is a critical area of research, given the constraints on energy consumption and the necessity for robust security measures. Recent studies have proposed various approaches to address these challenges.

- In 2019, a study by Zhang et al. [131] introduced a data fusion algorithm for wireless sensor networks inspired by hesitant fuzzy entropy. This approach effectively decreased data redundancy, reduced data transmission, and minimized energy consumption, extending the network's lifecycle and improving bandwidth utilization.

- In 2023, Neto et al. [132] proposed a data fusion and transfer learning empowered granular trust evaluation for the Internet of Things. This method aimed to solve issues related to confusing storage and insufficient fusion computing performance of multi- source heterogeneous distribution data.

- In 2023, Mahlake et al. [133] developed a hybrid algorithm to enhance wireless sensor network security within the IoT framework. This algorithm combined the Secure IoT (SIT) algorithm with the Security Protocols for Sensor Networks (SPINS) to create a Lightweight Security Algorithm (LSA), addressing data security concerns while reducing power consumption without sacrificing performance.

- Another 2023 study by Höglund and Raza [134] introduced BLEND, which combines secure storage and communication by storing IoT data as pre-computed encrypted network packets. This method eliminates the need for separate cryptographic protocol

For storage and communication, significantly reducing communication latency and local processing and enhancing energy efficiency.

- These studies collectively contribute to developing secure, low-energy data fusion algorithms in IoT networks, addressing energy efficiency and security concerns through innovative approaches.

- Designing an integrity verification protocol based on privacy-preserving homomorphic data fusion in IoT networks is a critical area of research, focusing on ensuring data integrity while maintaining confidentiality. Recent studies have explored various approaches to address these challenges.

- In 2022, Zhang et al. [135] developed an External Data Integrity Tracking and Verification System tailored for stream computing in IoT environments. This system addresses IoT data streams' real-time and volatile nature, providing a framework to ensure data integrity in such dynamic settings.

- In 2023, Alghazwi et al. [136] introduced VPAS, a protocol for publicly verifiable and privacy-preserving aggregate statistics on distributed datasets. VPAS ensures data integrity and privacy by utilizing homomorphic encryption and zero-knowledge proofs, making it suitable for IoT applications where data is distributed across multiple sources.

- In 2022, Kumar and Singh [137] proposed a Private Data Storage model for IoT data in cloud-fog computing environments. This model employs homomorphic encryption to partition and encrypt data at the edge, ensuring privacy preservation and secure aggregation in outsourced cloud settings.

- In 2021, an innovative data integrity verification scheme was proposed for IoT-assisted information exchange in transportation systems. This scheme addresses the need for credibility guarantees to prevent unauthorized changes, ensuring source validity and safeguarding sensitive information in vehicular networks.

These studies collectively contribute to developing integrity verification protocols that leverage privacy-preserving homomorphic data fusion techniques in IoT networks, addressing data integrity and confidentiality concerns through innovative approaches.

13. Conclusion

IDSs are a fundamental component of cybersecurity, with HIDS and NIDS providing tailored security solutions for host-based and network-based environments, respectively. Anomaly-based detection techniques, such as PCA, enable efficient and scalable threat detection, particularly in IoT and resource-constrained environments. However, misuse-based IDSs, despite their effectiveness in recognizing known attacks, require extensive updates and may struggle with performance limitations in memory- constrained networks. The ongoing evolution of IDS methodologies continues to enhance their ability to counter emerging cybersecurity threats.

References

Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion, 58*, 82–115. https://doi.org/10.1016/j.inffus.2019.12.012

da Costa, K. A. P., Papa, J. P., Lisboa, C. O., Munoz, R., & de Albuquerque, V. H. C. (2019). Internet of Things: A survey on machine learning-based intrusion detection approaches. *Computer Networks, 151*, 147–157. https://doi.org/10.1016/j.comnet.2019.01.023

García-Magariño, I., Muttukrishnan, R., & Lloret, J. (2019). Human-centric AI for trustworthy IoT systems with explainable multilayer perceptrons. *IEEE Access, 7*, 125562–125574. https://doi.org/10.1109/ACCESS.2019.2937521

Mane, S., & Rao, D. (2021). *Explaining network intrusion detection system using explainable AI framework*. arXiv. https://doi.org/10.48550/arXiv.2103.07110

Marino, D. L., Wickramasinghe, C. S., & Manic, M. (2018). An adversarial approach for explainable AI in intrusion detection systems. In *2018 44th Annual Conference of the IEEE Industrial Electronics Society (IECON)* (pp. 4183–4188). IEEE. https://doi.org/10.1109/IECON.2018.8592143

Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *2015 Military Communications and Information Systems Conference (MilCIS)* (pp. 1–6). IEEE. https://doi.org/10.1109/MilCIS.2015.7348942

Moustafa, N., Slay, J., & Turnbull, B. (2018). An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of Internet of Things. *IEEE Internet of Things Journal, 6*(3), 4815–4830. https://doi.org/10.1109/JIOT.2018.2875467

Wang, M., Wang, Z., & Ye, X. (2020). An explainable machine learning framework for intrusion detection systems. *IEEE Access, 8*, 73127–73141. https://doi.org/10.1109/ACCESS.2020.2988850

Wang, Z. (2018). Deep learning-based intrusion detection with adversaries. *IEEE Access, 6*, 38367–38384. https://doi.org/10.1109/ACCESS.2018.2852765

Zoghi, Z., & Serpen, G. (2021). *UNSW-NB15 computer security dataset: Analysis through visualization*. arXiv. https://doi.org/10.48550/arXiv.2101.05067