Enhancing IoT Smart Home Security Through Machine Learning-Based Cyberattack Detection: A Comparative Evaluation

Nurin Abyana Balqis Jailani[1], Haniza Nahar[2*], Nurhashikin Mohd Salleh[3], Zulkiflee Muslim[4]

[1,2]Faculty of Information & Communication Technology, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia

[3,4]Faculty of Artificial Intelligence & Cyber Security, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia

***Corresponding Author**

DOI: https://doi.org/10.47772/IJRISS.2026.100700159

Received: 14 July 2026; Accepted: 20 July 2026; Published: 27 July 2026

ABSTRACT

Smart homes depend on interconnected sensors, cameras, routers, mobile applications, and cloud services. This connectivity improves automation and convenience, but it also expands the attack surface for Distributed Denial of Service (DDoS), Denial of Service (DoS), Mirai botnet, brute-force, spoofing, reconnaissance, and man-in-the-middle attacks. Traditional signature-based security is often insufficient because IoT devices are resource-constrained, heterogeneous, and frequently deployed with weak authentication or delayed firmware updates. This study evaluates supervised machine-learning classifiers for detecting cyberattacks in smart-home IoT network traffic using the CICIoT2023 dataset. Four algorithms, namely Random Forest, Decision Tree, k-Nearest Neighbour, and Support Vector Machine, were compared under 50:50, 70:30, and 80:20 train-test split settings. The models were evaluated using accuracy, precision, recall, and F1-score, with emphasis on DDoS, Mirai, and brute-force attack classes that are particularly relevant to smart-home environments. The findings show that tree-based classifiers are highly effective for IoT attack detection. Random Forest achieved the strongest overall accuracy and precision, while Decision Tree showed the most stable recall and F1-score for brute-force detection. The results indicate that Random Forest is suitable as a general-purpose smart-home IDS classifier, whereas Decision Tree or a hybrid ensemble strategy should be considered when missed brute-force attacks carry high operational risk. The paper contributes a clearer empirical comparison of lightweight supervised learning models and provides implementation guidance for smart-home intrusion detection systems.

Keywords: Smart Home Cybersecurity, Machine Learning-Based Intrusion Detection, IoT Attack Detection, Network Traffic Analysis, CICIoT2023 Dataset.

INTRODUCTION

The Internet of Things (IoT) has become a major foundation of digital transformation because it enables physical devices to sense, collect, process, and exchange data through network connectivity. In smart homes, IoT technologies are embedded in routers, cameras, voice assistants, smart locks, sensors, appliances, lighting systems, and energy-management devices. These devices support remote monitoring and automation, but their continuous connectivity also creates persistent security exposure.

Smart-home IoT security is challenging for three related reasons. First, devices are heterogeneous and often rely on different communication protocols, firmware versions, and cloud services. Second, many devices have limited processing power and memory, which restricts the use of heavy cryptographic or host-based protection mechanisms. Third, consumers frequently deploy devices with default passwords, outdated firmware, and weak network segmentation. As a result, a compromised device can become an entry point for lateral movement, botnet recruitment, data theft, or service disruption.

DDoS, DoS, Mirai botnet, and brute-force attacks are especially important in smart-home environments. DDoS and DoS attacks degrade availability by flooding devices or gateways with excessive traffic. Mirai-style botnets exploit weak credentials and vulnerable services to recruit IoT devices into remotely controlled attack networks. Brute-force attacks target authentication mechanisms by repeatedly testing username-password combinations. These threats are difficult to manage using rule-based monitoring alone because attack traffic may vary across devices, protocols, and network conditions.

Machine learning (ML) provides a promising approach for intrusion detection because it can learn behavioural patterns from network traffic and classify unknown traffic based on extracted features. Recent datasets such as CICIoT2023 provide large-scale and realistic traffic traces that allow researchers to evaluate ML models against diverse IoT attack classes [1]. However, high accuracy alone is not sufficient for deployment. A model must also maintain strong precision, recall, and F1-score across attack types, especially when false negatives may allow persistent compromise.

This study therefore aims to evaluate and compare four supervised ML algorithms for detecting cyberattacks in smart-home IoT traffic. The specific objectives are: (i) to examine the detection performance of Random Forest, Decision Tree, k-Nearest Neighbour, and Support Vector Machine using CICIoT2023; (ii) to compare model performance across different train-test split ratios; and (iii) to identify the most suitable classifier for smart-home cyberattack detection based on accuracy, precision, recall, and F1-score. The remainder of this paper is organized as follows: the literature review, methodology, results and discussion, conclusion, acknowledgement, and references.

To better understand the current state of IoT intrusion detection research, the following section reviews recent studies on IoT cyberattacks, publicly available datasets, machine learning techniques, and existing research gaps.

LITERATURE REVIEW

The literature on IoT intrusion detection has grown rapidly between 2020 and 2026. Recent studies can be grouped into four main areas: IoT threat characteristics, public benchmark datasets, machine-learning and deep-learning detection models, and deployment issues such as feature selection, energy consumption, explainability, and federated learning. Together, these works show that ML-based intrusion detection is suitable for IoT security, but model selection must consider class-specific performance rather than relying only on aggregate accuracy.

IoT Attack Types

IoT networks are exposed to both conventional network attacks and device-specific attacks. DDoS and DoS attacks remain among the most damaging threats because they directly affect availability, especially for smart-home gateways and cameras. Mirai botnet attacks are particularly relevant because they exploit weak or default credentials, convert devices into bots, and coordinate large-scale malicious traffic. Brute-force attacks target authentication services and may become a precursor to botnet infection, privilege abuse, or unauthorized device control. Spoofing and man-in-the-middle attacks [4] compromise identity and traffic integrity, while reconnaissance allows attackers to map vulnerable devices before launching more destructive activity. The major cyberattacks commonly found in smart-home IoT environments are summarized in Table 1.

Table 1: Cyberattack in IoT environments

Attack type	Behaviour in smart-home IoT	Security impact
DDoS and DoS	Floods a device, gateway, or service with excessive traffic or requests.	Loss of availability, delayed automation, device instability, and possible service outage.
Mirai botnet	Compromises vulnerable IoT devices, commonly through weak credentials, and uses them as remotely controlled bots.	Large-scale attack participation, bandwidth abuse, device slowdown, and reputational or legal risk.

Brute force	Repeatedly attempts username-password combinations against device or service authentication.	Unauthorized access, credential compromise, and preparation for botnet infection.
Spoofing	Impersonate a legitimate device, user, or network entity.	False commands, trust violation, and possible bypass of access control.
Reconnaissance	Scans the network to identify devices, open ports, services, and vulnerabilities.	Attack preparation and exposure of weak devices or services.
Man-in-the-middle (MitM)	Intercepts or manipulates communication between legitimate endpoints.	Data leakage, command manipulation, and loss of confidentiality and integrity.

The attack types discussed above introduce several practical challenges when designing an intrusion detection system for smart-home environments.

IoT Security Challenges

Smart-home IoT security differs from traditional enterprise security because devices are usually small, diverse, and continuously connected. Many devices are deployed by non-specialist users, which increases the likelihood of weak passwords, unpatched firmware, and insecure network configurations. Security monitoring is also complicated by heterogeneous traffic patterns: a camera, smart bulb, thermostat, and home router may generate very different volumes, protocols, and timing patterns. In addition to conventional network traffic analysis, recent studies have explored the integration of side-channel information with machine learning techniques to improve intrusion detection accuracy. Campos et al. [8] showed that combining multiple sources of information enables IDS models to detect sophisticated attacks more effectively in heterogeneous IoT environments. Therefore, IDS models must handle diversity without becoming overly sensitive to benign variation.

Dataset quality is another major challenge. Older intrusion detection datasets are useful for baseline experiments but may not represent modern IoT behaviour, botnet patterns, or smart-home traffic. Newer datasets such as CICIoT2023 and Edge-IIoTset provide richer attack coverage and more realistic IoT/IIoT topologies [1] [10]. However, no dataset perfectly represents every deployment. For this reason, models should be evaluated using multiple performance metrics and, where possible, validated under different data splits and attack classes.

Machine Learning in Intrusion Detection

Supervised ML models are widely used in IoT IDS because they can classify network traffic after learning from labelled examples. Decision Tree and Random Forest models are attractive because they handle nonlinear feature interactions, support multiclass classification, and offer relatively interpretable decision structures. Random Forest reduces overfitting by combining multiple decision trees, often producing strong generalization performance. k-Nearest Neighbour can perform well when feature spaces are meaningful but may be computationally expensive for large datasets. Support Vector Machine can produce strong decision boundaries, but its performance and runtime depend heavily on feature scaling, kernel selection, and class distribution.

Recent research also explores deep learning, federated learning, and hybrid IDS approaches. Deep-learning models such as CNN, LSTM, and hybrid architectures can capture complex spatial and temporal patterns in network traffic [7, 14, 17]. Federated learning is increasingly studied because it allows distributed smart-home or edge devices to learn collaboratively without sharing raw traffic data [6, 18]. Nevertheless, lightweight classical ML remains important for practical smart-home IDS because it is easier to deploy, tune, interpret, and run on limited hardware.

Recent studies have also investigated lightweight machine learning models to improve intrusion detection efficiency while reducing computational overhead. For example, Altamimi and Abu Al-Haija [5] demonstrated

that efficient learning algorithms can achieve competitive detection performance with lower training complexity, making them suitable for resource-constrained IoT environments such as smart homes.

Comparative Analysis of Existing Works

Existing works show a consistent trend: tree-based and ensemble classifiers often perform strongly in IoT IDS tasks, especially when combined with feature selection and careful preprocessing. Studies using recent IoT datasets report high detection performance for ensemble models, while also noting that class imbalance and attack-specific variability can reduce recall for minority or subtle attacks [8, 10, 11]. Deep-learning approaches can improve representation learning but may require more computational resources and larger training data. Hybrid intrusion detection frameworks that combine machine learning and deep learning have also shown promising results. Shahid et al. [17] reported that integrating multiple learning techniques can improve detection capability by leveraging both traditional feature learning and deep neural representations, although this often increases computational requirements. Energy-aware studies further demonstrate that model selection should consider computational and energy costs in addition to predictive accuracy [19]. Table 2 summarizes representative studies related to machine learning-based intrusion detection in IoT environments.

Table 2: Several studies towards the role of Machine Learning in IoT Security

Study focus	Sources	Relevance to this study
Modern IoT datasets	[1, 10]	Supports the use of realistic and large-scale IoT traffic for IDS benchmarking.
DDoS and botnet detection	[2, 3, 14]	Shows the importance of attack-specific evaluation for availability-focused threats.
Classical and lightweight ML	[4, 13, 15]	Supports comparison of deployable supervised classifiers for IoT environments.
Feature selection and optimization	[11, 12, 20]	Highlights the need to reduce redundant features and improve class-specific detection.
Federated and smart-home IDS	[6, 16, 18]	Shows emerging directions for privacy-preserving and smart-home-specific IDS deployment.

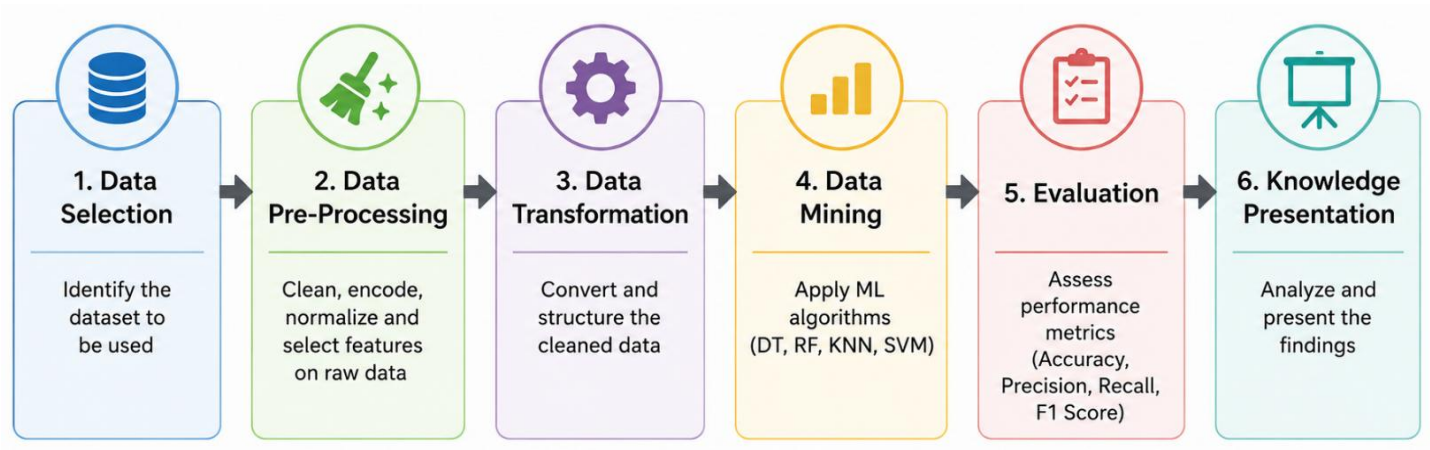
Research Gaps

Although many studies report high IDS accuracy, several gaps remain. First, aggregate accuracy can hide weak recall for specific attacks such as brute-force, where missed detections are operationally serious. Second, smart-home IDS research still needs clearer comparisons of lightweight models under consistent experimental settings. Third, many studies emphasize new architectures but provide limited practical guidance on when a simpler classifier is sufficient. Fourth, the literature increasingly recommends feature selection, explainability, and federated learning, but these approaches must be balanced against implementation complexity in consumer environments. This study addresses part of this gap by comparing four supervised models on CICIoT2023 using multiple split ratios and by discussing model suitability based on both overall and attack-specific performance.

METHODOLOGY

The study follows the Knowledge Discovery in Databases (KDD) process as shown in Figure 1 because it provides a systematic flow for dataset selection, preprocessing, transformation, data mining, evaluation, and interpretation. This structure is suitable for IDS experiments because it reduces overlap between preparation and modelling activities and ensures that performance results are traceable to a defined experimental procedure.

Figure 1: KDD Process



Dataset Description

This study uses the UNB CICIoT2023 dataset [1], a recent and diverse benchmark for IoT intrusion detection. The dataset contains traffic generated from a large IoT topology with 105 devices and 33 attack behaviours. CICIoT2023 includes seven broad attack categories: DDoS, DoS, reconnaissance, web-based attacks, brute-force, spoofing, and Mirai botnets [1]. Because the present study focuses on smart-home security, the analysis emphasizes DDoS, Mirai, and brute-force attacks, which are highly relevant to home gateways, cameras, routers, and weakly protected device interfaces.

Experimental Setup

The experiment was conducted in Python using standard ML libraries. The models were trained and evaluated using the same dataset partitions to ensure a fair comparison. The hardware and software environment is summarized in Table 3.

Table 3: Hardware and Software Requirements

Component	Specification or role
Processor	13th Gen Intel Core i5-13420H, 8 cores, 12 logical processors.
Memory and storage	16 GB RAM and 477 GB storage.
Operating system	Windows 11, 64-bit operating system.
Python	Used for model development and experiment execution.
Scikit-learn	Used to implement Random Forest, Decision Tree, k-NN, SVM, train-test splits, and performance metrics.
Matplotlib and Seaborn	Used to visualize model performance across attack classes and split ratios.

Machine Learning Models

Random Forest was selected because it is an ensemble classifier that reduces variance by aggregating multiple decision trees. Decision Tree was selected as a lightweight and interpretable baseline. k-Nearest Neighbour was included because it classifies samples based on proximity in feature space and is often used as a non-parametric baseline. Support Vector Machine was selected because it can construct separating hyperplanes for complex classification tasks, although its performance may be sensitive to feature scaling, kernel selection, and class distribution.

Performance Metrics

The performance of each classifier was evaluated using a Confusion Matrix as in Table 4 and four (4) metrics: accuracy, precision, recall, and F1-score. The detailed explanation for each metric can be found in Table 5.

Table 4: Confusion Matrix

	Predicted Positive	Predicted Negative
Actual True	True Positive (TP)	True Negative (TN)
Actual False	False Positive (FP)	False Negative (FN)

Table 5: Performance Metrics

Metric & Formula	Interpretation
$\boldsymbol{Accuracy}, Acc = \frac{\text{Correct Predictions}}{\text{Total Predictions}}$ $Acc = \frac{TN+TP}{TP+FP+TN+FN}$	The rate of correct classification, regardless of the class. It measures how effectively a classification model predicts the actual results or labels of a dataset.
$\boldsymbol{Precision}, P = \frac{\text{The Positive Predicted}}{\text{Total Positive Prediction}}$ $P = \frac{TP}{TF+TP}$	Reliability of correct positive predictions generated by a classification model.
$\boldsymbol{Recall}, R = \frac{\text{The Positive Predicted}}{\text{Total Positive and Negative Prediction}}$ $R = \frac{TP}{FN+TP}$	The efficiency of the classifier to detect the correct class. It quantifies how well a classification model can locate and categorize every positive instance presented in a dataset.
$\boldsymbol{F1\ Score}, F = \frac{2 \text{ x } Precision \text{ x } Recall}{Precision+Recall}$	The F1-score is the harmonic mean of precision and recall. It provides a balanced assessment by giving greater weight to the lower of the two values and ranges from 0 to 1.

RESULTS AND DISCUSSION

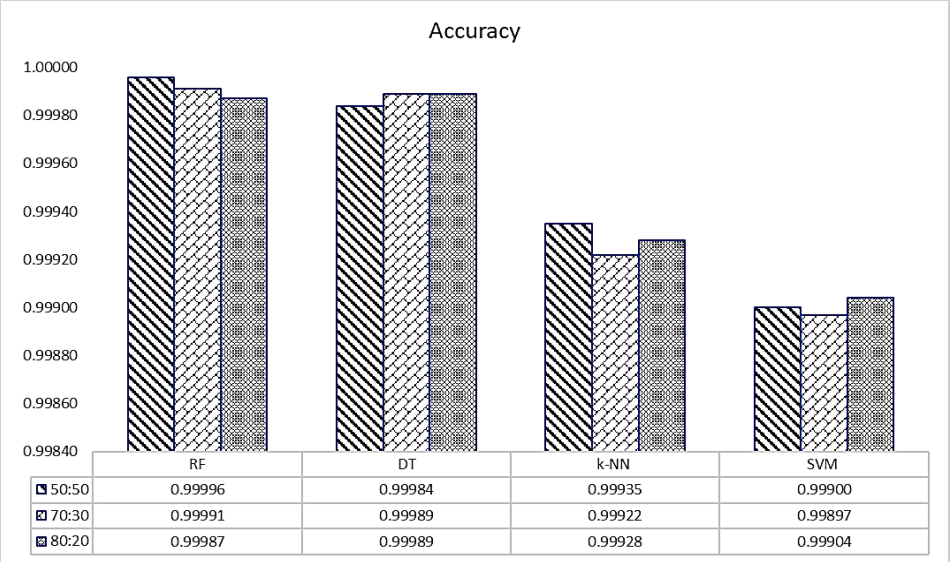
Four supervised ML models were evaluated across three train-test split ratios: 50:50, 70:30, and 80:20. The selected models were assessed using accuracy, precision, recall, and F1-score. The goal was to identify the classifier that provides the most reliable detection of IoT-based cyberattacks in smart-home network traffic.

Accuracy

Accuracy results show that all four models achieved high overall detection performance as shown in Figure 2. Random Forest produced the highest and most consistent accuracy across the three split ratios, with an average accuracy of approximately 99.991%. Decision Tree also performed strongly, with an average accuracy of approximately 99.987%. k-NN and SVM achieved lower values than the tree-based models but still exceeded 99.897%. These results confirm that CICIoT2023 traffic contains discriminative features that supervised classifiers can learn effectively.

However, the small differences in aggregate accuracy should be interpreted carefully because they may hide important attack-specific weaknesses. The strong performance of the tree-based classifiers is consistent with previous studies showing that Random Forest and related ensemble approaches can provide robust generalization for IoT intrusion detection [13, 15].

Figure 2: Accuracy values of cyberattacks across different ML Models

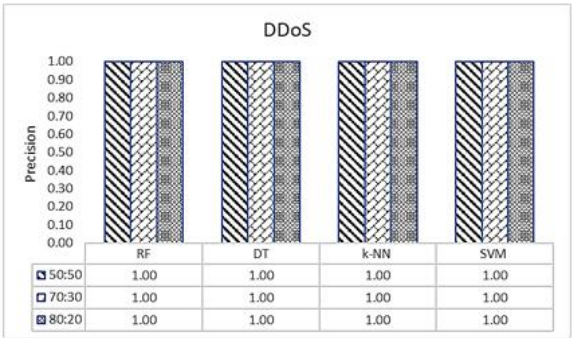


Precision

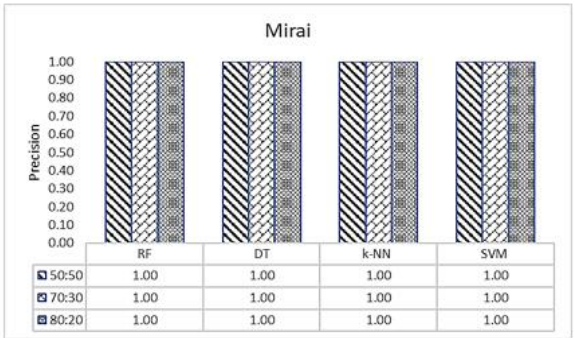
Precision results indicate that the evaluated models were generally reliable when predicting malicious traffic as shown in Figure 3. DDoS and Mirai attacks were classified with near-perfect or perfect precision by all models across the tested split ratios. This suggests that these attack classes have clear traffic characteristics in the selected feature space. For brute-force attacks, Random Forest maintained the strongest precision among the tested classifiers, indicating that its positive brute-force predictions were less likely to be false alarms. High precision is useful for smart-home IDS deployment because users and administrators may ignore alerts if the system produces excessive false positives.

The near-perfect precision for DDoS and Mirai detection is consistent with studies reporting that these attacks generally exhibit highly distinguishable traffic patterns when appropriate features are selected [12, 14].

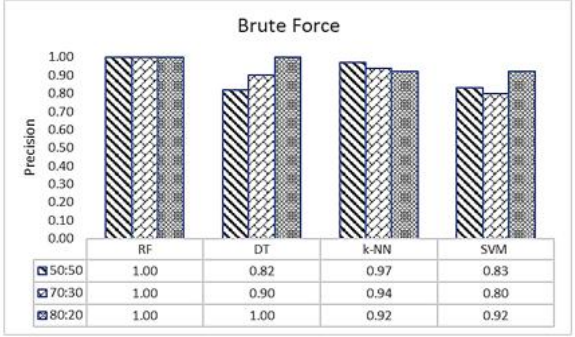
Figure 3: Precision values on (a) DDoS (b) Mirai (c) Brute Force detection across different ML Models



(a)



(b)



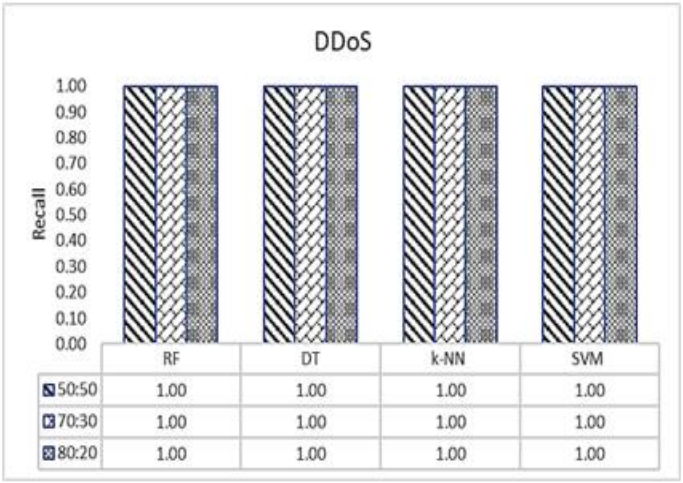
(c)

Recall

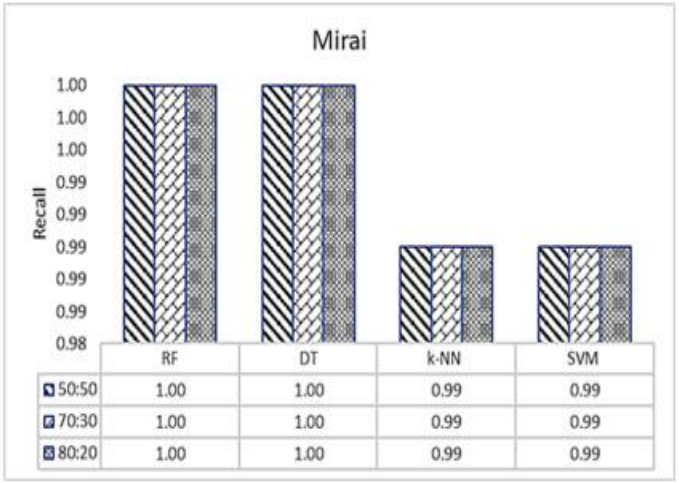
Recall provides a stricter view of IDS effectiveness because it measures whether actual attacks are detected. All models correctly detected DDoS attacks in the evaluated dataset as shown in Figure 4. For Mirai attacks, Random Forest and Decision Tree consistently achieved recall of 1.00, while k-NN and SVM produced slightly lower but still strong recall values. Brute-force detection was more difficult. Decision Tree achieved the most stable recall for brute-force attacks, while Random Forest and k-NN varied between approximately 0.69 and 0.90. SVM showed the weakest brute-force recall, ranging approximately from 0.10 to 0.16. This finding is important because a model with high accuracy but low brute-force recall may still miss unauthorized login attempts.

The variation in brute-force recall further demonstrates that high aggregate accuracy does not necessarily indicate reliable detection across all attack classes, particularly for attacks with less distinctive traffic characteristics [9, 11].

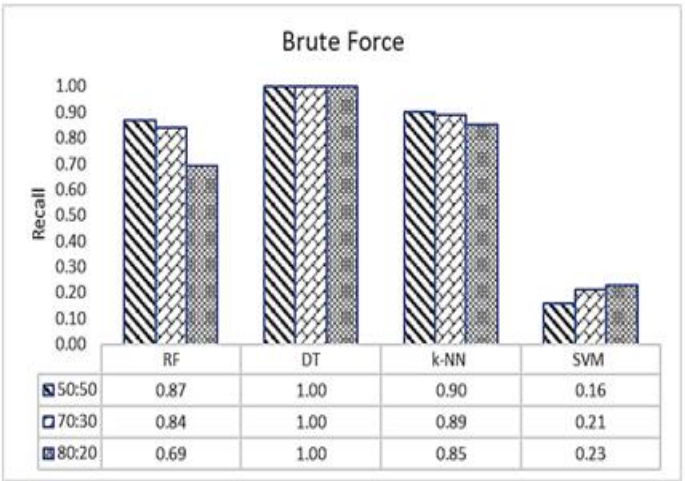
Figure 4: Recall values on (a) DDoS (b) Mirai (c) Brute Force detection across different ML Models



(a)



(b)



(c)

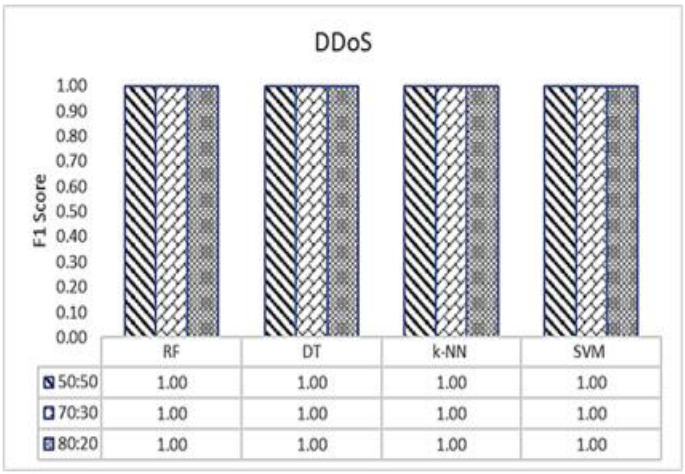
F1-score

The F1-score results confirm the trade-off between precision and recall as in Figure 5. For DDoS attacks, all models achieved an F1-score of 1.00. For Mirai attacks, Random Forest and Decision Tree consistently achieved an F1-score of 1.00, while k-NN and SVM varied slightly between 0.99 and 1.00. For brute-force attacks, Decision Tree showed the most stable balanced performance. Random Forest and k-NN achieved moderate to strong F1-scores, while SVM achieved weak values of approximately 0.27 to 0.35 because of its low recall. These

results suggest that SVM is less suitable for brute-force detection under the current configuration and feature representation.

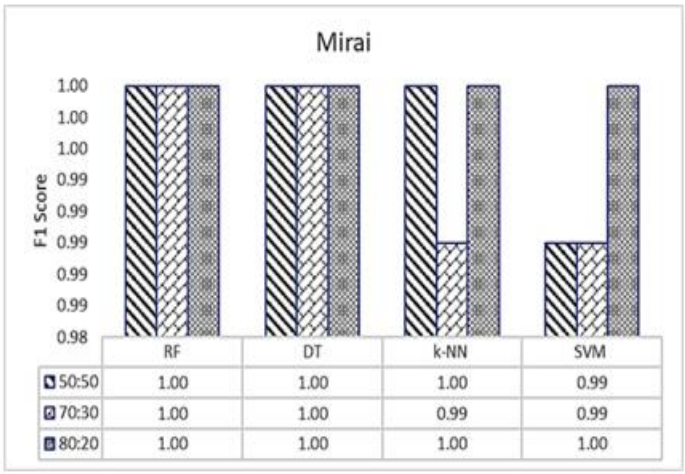
The weak brute-force F1-score obtained by SVM suggests that model suitability depends not only on the classifier but also on feature scaling, class representation, and hyperparameter configuration.

Figure 5: F1-score values on (a) DDoS (b) Mirai (c) Brute Force detection across different ML Models



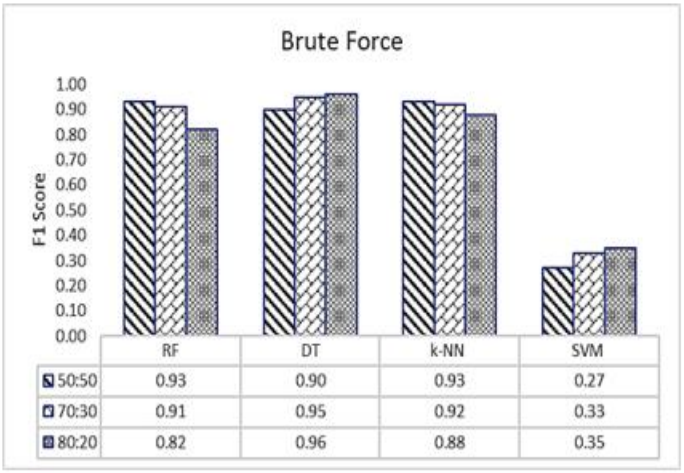
	RF	DT	k-NN	SVM
50:50	1.00	1.00	1.00	1.00
70:30	1.00	1.00	1.00	1.00
80:20	1.00	1.00	1.00	1.00

(a)



	RF	DT	k-NN	SVM
50:50	1.00	1.00	1.00	0.99
70:30	1.00	1.00	0.99	0.99
80:20	1.00	1.00	1.00	1.00

(b)



	RF	DT	k-NN	SVM
50:50	0.93	0.90	0.93	0.27
70:30	0.91	0.95	0.92	0.33
80:20	0.82	0.96	0.88	0.35

(c)

The individual evaluation metrics discussed above provide a detailed understanding of each classifier. An overall comparison is presented in the following section to summarize their practical suitability.

Overall Models Comparison

Overall, Random Forest is selected as the most suitable general classifier because it achieved the strongest average accuracy and maintained consistently high precision across the evaluated attacks and split ratios. Nevertheless, the results also show that Decision Tree is more stable for brute-force recall and F1-score. Therefore, a practical smart-home IDS could deploy Random Forest as the primary classifier while using Decision Tree or an ensemble decision rule to strengthen brute-force sensitivity. This recommendation is consistent with recent literature showing that ensemble and tree-based models often provide strong IoT IDS performance, especially when dataset features are well prepared [9, 13, 15].

Figure 6 summarizes the overall performance comparison across all evaluated classifiers and train–test split ratios. The comparison indicates that Random Forest consistently achieved the best balance between Accuracy, Precision, Recall, and F1-score.

Figure 6: Comparative performance evaluation of machine learning algorithms for cyberattack detection under different train–test split ratios (50:50, 80:20, and 70:30).

Ratio	Algorithm	Performance Metrics											
		Accuracy			Precision			Recall			F1 Score		
		DDoS	Mirai	Brute Force	DDoS	Mirai	Brute Force	DDoS	Mirai	Brute Force	DDoS	Mirai	Brute Force
50:50	RF	**0.99996 (BEST)**			1.00	1.00	1.00	1.00	1.00	0.87	1.00	1.00	0.93
	DT	0.99984			1.00	1.00	0.82	1.00	1.00	1.00	1.00	1.00	0.90
	k-NN	0.99935			1.00	1.00	0.97	1.00	0.99	0.90	1.00	1.00	0.93
	SVM	0.999			1.00	1.00	0.83	1.00	0.99	0.16	1.00	0.99	0.27
80:20	RF	0.99987			1.00	1.00	1.00	1.00	1.00	0.69	1.00	1.00	0.82
	DT	**0.99989 (BEST)**			1.00	1.00	0.93	1.00	1.00	1.00	1.00	1.00	0.96
	k-NN	0.99928			1.00	1.00	0.92	1.00	0.99	0.85	1.00	1.00	0.88
	SVM	0.99904			1.00	1.00	0.75	1.00	0.99	0.23	1.00	1.00	0.35
70:30	RF	**0.99991 (BEST)**			1.00	1.00	1.00	1.00	1.00	0.84	1.00	1.00	0.91
	DT	0.99989			1.00	1.00	0.90	1.00	1.00	1.00	1.00	1.00	0.95
	k-NN	0.99922			1.00	1.00	0.94	1.00	0.99	0.89	1.00	0.99	0.92
	SVM	0.99897			1.00	1.00	0.80	1.00	0.99	0.21	1.00	0.99	0.33

Error! Reference source not found. Therefore, the results of this study clearly highlight its significance in the following ways:

- Random Forest (RF) achieved the best overall performance, consistently providing the highest accuracy and balanced Precision, Recall, and F1-score across different train–test ratios. This finding is consistent with previous IoT intrusion detection studies reporting that tree-based and lightweight ensemble classifiers can provide strong predictive performance and generalization [13, 15].
- Decision Tree (DT) also performed very well, achieving the highest accuracy (99.989%) at the 80:20 split while maintaining strong detection performance.
- K-Nearest Neighbors (k-NN) showed stable performance, but its Recall and F1-score for brute-force attacks were slightly lower than RF and DT.
- Support Vector Machine (SVM) recorded the weakest performance, particularly in detecting brute-force attacks, despite having high overall accuracy. This shows that accuracy alone is not sufficient to evaluate intrusion detection models.
- Brute-force attacks were the most difficult to detect, while DDoS and Mirai attacks were classified almost perfectly by most algorithms.
- The train–test split ratio had little effect on model performance, indicating that the classifiers remained stable and generalized well across different data partitions.
- Overall, Random Forest is the most suitable model for IoT cyberattack detection, offering the most reliable and consistent performance across all evaluation metrics.

Subsequently,

Table **6** highlights the strengths and limitations of each model and provides recommendations for their practical deployment in real-world cyberattack detection systems.

Table 6: Summary on ML models – Pro vs. Cons

Model	Observed strength	Observed limitation	Practical implication
Random Forest	Highest overall accuracy and consistently strong precision.	Brute-force recall varied under some split ratios.	Recommended as the best general-purpose classifier for smart-home IDS.

Decision Tree	Very stable recall and F1-score, especially for brute-force attacks.	Slightly lower average accuracy than Random Forest.	Suitable when interpretability and attack recall are prioritized.
k-Nearest Neighbour	High overall accuracy and acceptable performance for major attack classes.	Performance may decline with high-dimensional or large datasets.	Useful baseline, but less attractive for real-time deployment.
Support Vector Machine	Can classify clear attack patterns such as DDoS.	Weak brute-force recall and F1-score in this experiment.	Requires further tuning before deployment for smart-home IDS.

Based on the comparison presented in

Table **6**, each machine learning model offers distinct advantages depending on deployment requirements, computational resources, and detection priorities.

Implication for Smart-Home Security

The findings demonstrate that ML-based IDS can enhance smart-home security by identifying malicious network patterns that may not be captured by static rules. High DDoS and Mirai detection performance is valuable because smart-home devices are often recruited into botnets or targeted for availability disruption. However, brute-force detection requires special attention because missed login attempts may lead to unauthorized access. A deployment-ready IDS should therefore combine model performance with secure device configuration, strong password policies, firmware updates, network segmentation, and continuous monitoring.

CONCLUSION

This study evaluated Random Forest, Decision Tree, k-Nearest Neighbour, and Support Vector Machine for machine-learning-based cyberattack detection in smart-home IoT networks using the CICIoT2023 dataset. The results show that tree-based classifiers are especially effective for IoT intrusion detection. Random Forest achieved the best overall accuracy and precision, making it the most suitable general-purpose classifier among the models tested. Decision Tree showed the most stable recall and F1-score for brute-force attacks, indicating that it is valuable when the detection of login-based attacks is a priority.

The study contributes a more comprehensive comparison of supervised ML models for smart-home cyberattack detection and highlights the importance of evaluating attack-specific recall and F1-score rather than relying only on aggregate accuracy. The findings support the use of ML-based IDS as part of a broader smart-home security strategy. However, the results should be interpreted within the limits of the dataset and experimental design. Future work should evaluate additional algorithms such as gradient boosting, XGBoost, CNN–LSTM, and federated learning approaches [6, 18], apply feature-selection methods to reduce computational and deployment costs [11, 20], test the models on edge devices; and validate their performance using live or cross-dataset smart-home traffic. Future studies should also examine explainability so that IDS alerts can be trusted and acted upon by users and security administrators.

Future research should therefore focus on validating intrusion detection models across multiple IoT cybersecurity datasets and real-world smart-home deployments to improve model robustness and generalisability. Comparative investigations involving lightweight machine learning, deep learning, federated learning, and anomaly-based detection techniques are expected to provide deeper insights into their capability for detecting both known and zero-day cyberattacks. Furthermore, practical deployment studies should evaluate computational efficiency, inference latency, memory utilization, communication overhead, and energy consumption on resource-constrained edge devices. Incorporating explainable artificial intelligence and adaptive learning mechanisms will further enhance the transparency, reliability, and long-term effectiveness of intelligent intrusion detection systems for next-generation smart-home environments.

ACKNOWLEDGEMENT

The authors would like to thank the Research Group of Information Security Forensics and Computer Networking (INSFORNET), Center for Advanced Computing Technology (C-ACT), Fakulti Teknologi Maklumat dan Komunikasi (FTMK), Universiti Teknikal Malaysia Melaka for the financial support through PJP/2024/FTMK/PERINTIS/SA0038.

REFERENCES

1. Neto, E. C. P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R., & Ghorbani, A. A. (2023). CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment. Sensors, 23(13), Article 5941. https://doi.org/10.3390/s23135941
2. Alani, M. M. (2022). BotStop: Packet-based efficient and explainable IoT botnet detection using machine learning. Computer Communications, 193, 53-62. https://doi.org/10.1016/j.comcom.2022.06.039
3. Akgun, D., Hizal, S., & Cavusoglu, U. (2022). A new DDoS attacks intrusion detection model based on deep learning for cybersecurity. Computers & Security, 118, Article 102748. https://doi.org/10.1016/j.cose.2022.102748
4. Ali, M. A., & Al-Sharafi, S. A. H. (2025). Intrusion detection in IoT networks using machine learning and deep learning approaches for MitM attack mitigation. Discover Internet of Things, 5(1), Article 48. https://doi.org/10.1007/s43926-025-00104-w
5. Altamimi, S., & Abu Al-Haija, Q. (2024). Maximizing intrusion detection efficiency for IoT networks using extreme learning machine. Discover Internet of Things, 4(1), Article 5. https://doi.org/10.1007/s43926-024-00060-x
6. Attota, D. C., Mothukuri, V., Parizi, R. M., & Pouriyeh, S. (2021). An ensemble multi-view federated learning intrusion detection for IoT. IEEE Access, 9, 117734-117745. https://doi.org/10.1109/ACCESS.2021.3107337
7. Awajan, A. (2023). A novel deep learning-based intrusion detection system for IoT networks. Computers, 12(2), Article 34. https://doi.org/10.3390/computers12020034
8. Campos, A. D., Lemus-Prieto, F., Gonzalez-Sanchez, J.-L., & Lindo, A. C. (2024). Intrusion detection for IoT environments through side-channel and machine learning techniques. IEEE Access, 12, 98450-98465. https://doi.org/10.1109/ACCESS.2024.3362670
9. Fares, H., Aknin, N., Lazrek, G., & Zeroual, M. (2025). Intrusion detection in IoT environment using hyperparameters tuned machine and deep learning models on the CICIoT2023 dataset. Informatica, 49(5). https://doi.org/10.31449/inf.v49i5.8881
10. Ferrag, M. A., Friha, O., Hamouda, D., Maglaras, L., & Janicke, H. (2022). Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning. IEEE Access, 10, 40281-40306. https://doi.org/10.1109/ACCESS.2022.3165809
11. Garcia, J., Entrena, J., & Alesanco, A. (2024). Empirical evaluation of feature selection methods for machine learning based intrusion detection in IoT scenarios. Internet of Things, 28, Article 101367. https://doi.org/10.1016/j.iot.2024.101367
12. Hajjouz, A., & Avksentieva, E. (2024). Optimizing intrusion detection for DoS, DDoS, and Mirai attacks subtypes using hierarchical feature selection and CatBoost on the CICIoT2023 dataset. Data and Metadata, 3, Article 577. https://doi.org/10.56294/dm2024577
13. Hakami, H., Faheem, M., & Bashir Ahmad, M. (2025). Machine learning techniques for enhanced intrusion detection in IoT security. IEEE Access, 13, 31140-31158. https://doi.org/10.1109/ACCESS.2025.3542227
14. Hizal, S., Cavusoglu, U., & Akgun, D. (2024). A novel deep learning-based intrusion detection system for IoT DDoS security. Internet of Things, 28, Article 101336. https://doi.org/10.1016/j.iot.2024.101336
15. Ismail, S., Dandan, S., & Qushou, A. (2025). Intrusion detection in IoT and IIoT: Comparing lightweight machine learning techniques using TON_IoT, WUSTL-IIOT-2021, and EdgeIIoTset datasets. IEEE Access, 13, 73468-73485. https://doi.org/10.1109/ACCESS.2025.3554083
16. Moinuddin, K., & Prabhavathi, S. (2026). An integrated machine learning and deep learning framework for intrusion detection in IoT smart homes. Discover Internet of Things. https://doi.org/10.1007/s43926-026-00390-y

17. Shahid, U., Zunnurain Hussain, M., Zulkifl Hasan, M., Haider, A., Ali, J., & Altaf, J. (2024). Hybrid intrusion detection system for RPL IoT networks using machine learning and deep learning. IEEE Access, 12, 113099-113112. https://doi.org/10.1109/ACCESS.2024.3442529
18. Shalan, M., Hasan, M. R., Bai, Y., & Li, J. (2025). Enhancing smart home security: Blockchain-enabled federated learning with knowledge distillation for intrusion detection. Smart Cities, 8(1), Article 35. https://doi.org/10.3390/smartcities8010035
19. Tekin, N., Acar, A., Aris, A., Uluagac, A. S., & Gungor, V. C. (2023). Energy consumption of on-device machine learning models for IoT intrusion detection. Internet of Things, 21, Article 100670. https://doi.org/10.1016/j.iot.2022.100670
20. Walling, S., & Lodh, S. (2024). Enhancing IoT intrusion detection through machine learning with AN-SFS: A novel approach to high performing adaptive feature selection. Discover Internet of Things, 4(1), Article 16. https://doi.org/10.1007/s43926-024-00074-5