2026 Volume: 02 Issue: 07 www.ijrpa.com ISSN 2456-9995 Research Article

International Journal Research Publication Analysis

Page: 01-24

# BEHAVIOUR-BASED MULTILAYER INTRUSION DETECTION IN MOBILE AD HOC NETWORKS: A COMPARATIVE EVALUATION OF K-MEANS AND SUPPORT VECTOR MACHINES

**[*1]Deepa P. Vaidya, [2](Prof.) Dr. S. P. Deshpande**

[1]Assistant Professor, P.G. Dept. of Computer Science & Technology, Degree College of Physical Education, Shree H. V. P. Mandal, Amravati.

[2]Principal, Degree College of Physical Education, Shree H. V. P. Mandal, Amravati.

**Article Received: 05 June 2026**
**Article Revised: 25 June 2026**
**Published on: 15 July 2026**

***Corresponding Author: Deepa P. Vaidya**
Assistant Professor, P.G. Dept. of Computer Science & Technology, Degree College of Physical Education, Shree H. V. P. Mandal, Amravati.

DOI: https://doi-doi.org/101555/ijrpa.1494

## ABSTRACT

Mobile Ad Hoc Networks (MANETs) support infrastructure-independent communication in disaster response, military, and intelligent transportation environments, but their mobility, decentralization, and limited node resources expose them to attacks at multiple protocol layers. This study compares an unsupervised K-Means intrusion detector with a supervised Support Vector Machine (SVM) using a radial basis function kernel. A Python-based framework combining SimPy and NetworkX modelled 20 mobile nodes under Random Waypoint mobility and implemented 12 attack scenarios spanning the range from the physical to the application layer. Packet drop rate, control packet rate, and link failure count were used as lightweight behavioural features. Performance was assessed over ten independent simulation runs using detection rate, false-positive rate, precision, F1-score, and accuracy. The SVM achieved an overall accuracy of

0.98 and an F1-score of 0.70, compared with 0.69 and 0.51, respectively, for K-Means. Both methods detected Jamming, Sinkhole, and SYN Flood effectively; however, topology- and identity-oriented attacks, particularly Wormhole, Hijacking, and Sybil, remained difficult to identify. The results indicate that supervised learning offers stronger discrimination under labelled conditions, whereas unsupervised clustering remains attractive when labels are unavailable. Richer temporal, routing, and cross-layer features are required for robust detection of stealthy attacks.

**KEYWORDS:** Mobile Ad Hoc Networks, Intrusion Detection System, Machine Learning, Support Vector Machine, K-Means Clustering, Network Security, Multi-layer attacks.

## 1. INTRODUCTION

The rapid development of wireless communication has expanded the use of Mobile Ad Hoc Networks (MANETs) in battlefield communication, disaster recovery, emergency response, intelligent transportation, and related infrastructure-limited environments [1]. Unlike conventional wireless networks, a MANET operates without fixed infrastructure or centralised control and can therefore be deployed rapidly where permanent communication facilities are unavailable, damaged, or economically impractical [2]. Each mobile node acts as both a host and a router, cooperatively forwarding packets through multi-hop paths while the network topology changes over time [1,2].

These advantages are accompanied by substantial security risks. A recent integrated MANET framework jointly addressed cluster-head selection and intrusion detection, illustrating the growing use of machine learning in resource-aware network management [3]. Node mobility, time-varying links, distributed routing, and the absence of continuously available trusted infrastructure increase the attack surface and complicate centralised monitoring [4]. The shared wireless medium also exposes communication to interception, interference, eavesdropping, and jamming, while limitations in battery power, bandwidth, processing capability, and memory restrict the use of computationally intensive defence mechanisms [2,5]. Recent optimisation-based and deep supervised intrusion-detection frameworks have improved detection performance, but their multiple processing stages may impose overhead that is difficult to sustain on resource-constrained nodes [6]. Packet-dropping behaviour remains particularly difficult to distinguish from losses caused by mobility and channel variability [7].

MANET attacks can be classified according to their mechanism, objective, and affected protocol layer. Blackhole, Greyhole, Sinkhole, Wormhole, Flooding, Sybil, and route-hijacking attacks primarily disrupt routing and forwarding at the network layer [8,9]. Jamming targets the physical channel, SYN Flood exhausts transport-layer connection state, Session Hijacking exploits session control, Decrypt Bypass represents a presentation-layer security circumvention scenario, and application-layer distributed denial-of-service (DDoS) attacks overwhelm specific services [10]. This diversity makes it difficult for a single detection strategy to identify both disruptive and stealthy behaviours.

Signature-based intrusion detection systems identify threats by matching observed activity against known attack patterns. They are effective for previously characterised attacks but are less suitable for zero-day threats, modified attack variants, and behaviours for which signatures do not yet exist. Anomaly-based systems address this limitation by modelling expected behaviour and identifying deviations [11,12]. Nevertheless, false alarms may increase when legitimate mobility produces patterns that resemble malicious activity, and results obtained in one network environment may not generalise directly to another [13]. Secure routing and transmission mechanisms can reduce attack exposure, but they do not eliminate the need for behavioural monitoring [14].

Machine learning provides both supervised and unsupervised mechanisms for behavioural intrusion detection. Classification models such as Support Vector Machines (SVMs) learn decision boundaries from labelled examples, whereas clustering methods such as K-Means group observations without requiring attack labels. Hybrid classification and fuzzy-clustering studies have reported strong results on complex network traffic [15], and broader reviews confirm the increasing use of machine learning and deep learning in cybersecurity [16]. However, the relative performance of supervised and unsupervised methods is often difficult to assess because studies use different datasets, features, attack sets, and evaluation procedures.

This study therefore compares K-Means clustering and an RBF-kernel SVM under the same MANET simulation conditions. Twelve attack scenarios distributed across the range from OSI Layer 1 to Layer 7 are represented through three lightweight behavioural features: packet drop rate, control packet rate, and link failure count. The principal contributions are: (i) a unified Python-based simulation and evaluation framework; (ii) an attack-wise comparison of supervised and unsupervised detection under identical feature conditions; and (iii) an analysis of the attacks for which the compact feature set succeeds or fails. The central hypothesis is that the supervised SVM will provide stronger overall discrimination when reliable labels are available, while K-Means will retain practical value in settings where labels cannot be obtained.

The remainder of this paper is organised as follows. Section 2 reviews related research. Section 3 describes the simulation, attack modelling, feature extraction, learning algorithms, and evaluation procedure. Section 4 reports the experimental results, discussion, and Section 5 concludes the paper.

## 2. LITERATURE REVIEW

Machine learning has become an important approach for intrusion detection in Mobile Ad

Hoc Networks (MANETs) because conventional security mechanisms are often inadequate for decentralized and highly dynamic wireless environments. Recent studies have focused on improving intrusion detection accuracy while maintaining the resource efficiency required by MANETs. In their work, researcher built a combined intrusion detection system based on Adaptive Ensemble Tree Learning (AETL) supported with a Machine Learning Model Hybrid Dual Optimization (HDOMLM) for the two most important operations in MANETs, i.e. efficient cluster head selection and attack detection [3]. Their system resulted in a great number of enhancements among which, 71% in energy savings and more than 50% in lifetime extension of the network compared to the existing ANFIS-EESC and CCCH methods are the most meaningful ones.

Hassan et al. [4] reviewed machine learning and optimization techniques for detecting Blackhole and Greyhole attacks in MANETs. Their survey highlighted the effectiveness of federated learning, long short-term memory (LSTM), reinforcement learning and metaheuristic optimization algorithms for improving intrusion detection accuracy and network resilience. However, the study also indicated that many of these techniques introduce considerable computational overhead, limiting their suitability for resource-constrained MANET nodes.

Sheela et al. [6] introduced the AOMA-DSLC structure which is a mix of the Adaptive Marine Predator Optimization Algorithm and Deep Supervised Learning Classification. It helps with intrusion detection by automatically determining the most pertinent characteristics and effectively training the model. These studies demonstrate that hybrid machine learning models can significantly improve intrusion detection performance. However, their computational complexity and dependence on multiple optimization stages may restrict their deployment in real-time MANET environments.

Several recent studies have compared clustering-based and classification-based intrusion detection techniques to evaluate their suitability for network security applications. Ahmed et al.

[15] proposed a signature-based intrusion detection framework that integrates machine learning, deep learning, and fuzzy clustering. Their study demonstrated that classifiers such as Support Vector Machine (SVM), Random Forest (RF), Decision Tree (DT), Long Short-Term Memory (LSTM), and Artificial Neural Network (ANN) effectively distinguish normal traffic from malicious traffic. The paper also indicated that the combination of fuzzy clustering and machine learning not only improves the accuracy of intrusion detection but also makes it possible to handle poorly defined scenarios and complex patterns of network traffic.

Xin et al. [19] presented a comprehensive review of the application of machine learning and deep learning techniques in cybersecurity. Their study examined the commonly used machine learning and deep learning algorithms for intrusion detection, discussed the benchmark datasets widely adopted for performance evaluation, and identified the major challenges associated with applying these techniques to cybersecurity problems. The authors also highlighted several promising research directions for developing more robust, adaptable, and intelligent intrusion detection systems. Their findings emphasize the importance of selecting appropriate machine learning techniques and continuously improving intrusion detection systems to address the evolving nature of cyber threats. Owing to the complexity of detecting multiple attack types in wireless ad hoc networks, a wide range of intrusion detection systems has been proposed to address different forms of malicious activities.

Jassim et al. [8] used NS2 simulations to study blackhole and denial-of-service (DoS) attacks on MANETs using AODV as the routing protocol. They were able to demonstrate the detection, removal and isolation of malicious nodes and evaluated network performance by four metrics - packet delivery ratio, end-to-end delay, routing overhead and residual energy.

Yazdanypoor et al. [17] came up with a hybrid detection method for countering blackhole and greyhole attacks, which is based on the combination of anomaly detection, data mining techniques and cryptographic verification. It is a multilayered defense system operation at the mechanism level. They have effectively proven their method through their intrusion detection results, i.e. having a lower number of false positives plus very high percentages of packet delivery even in the presence of attackers.

Federated learning and deep learning have changed the way intrusion detection can be done by eliminating the privacy and scalability issues of centralized systems. The authors of paper [18] developed the HIDE-MAN system with Coati-Optimized BiLSTM method that was incorporated with Federated Learning and GANs which resulted in detection rate increases of 18.9%, 18.07% and 4.03% over KBIDS, WOA-DNN and MSA-GCNN, respectively. The framework preserved data privacy by performing model training locally while exchanging only model parameters during global aggregation.

Zhu and Liu [19] suggested an IoT technique for detecting network intrusions utilizing subspace clustering and ensemble learning. Their system uses CLIQUE PROCLUS SUBCLU and LOF clustering algorithms with a Logistic Regression meta-learner and it has a 97.05% accuracy on the UNSW-NB15 dataset, which demonstrated the potential of combining clustering and ensemble learning for intrusion detection.

John et al. conducted a thorough review of the intrusion detection systems for cluster-based

wireless sensor networks (WSNs) [20]. The authors reviewed different machine learning methods and feature selection techniques. In addition, they employed bibliometric analysis to highlight the major research trends in this field and to identify several research gaps with scopes for future research with designing efficient intrusion detection systems for resource-constrained wireless environments.

The identified research gaps provide valuable directions for developing lightweight and energy-efficient intrusion detection systems suitable for resource-constrained MANET environments.

The literature reveals three limitations that motivate the present study. First, many studies focus on one attack family, particularly Blackhole and Greyhole attacks, rather than evaluating threats distributed across several protocol layers. Second, comparisons between supervised and unsupervised methods frequently use different feature sets or experimental conditions. Third, high-performing deep and hybrid models may incur computational and communication overhead that is unsuitable for lightweight deployment. The present work addresses these limitations by evaluating K-Means and SVM models using the same three behavioural features, the same twelve attack scenarios, and the same simulation framework.

## 3. MATERIALS AND METHODS

The experimental workflow integrates discrete-event simulation, dynamic graph modelling, attack injection, behavioural feature extraction, machine learning, and metric-based evaluation, as summarised in Figure 1. The framework is intended as a controlled behavioural abstraction for comparing detection algorithms; it does not reproduce every operation of a complete physical and protocol stack.

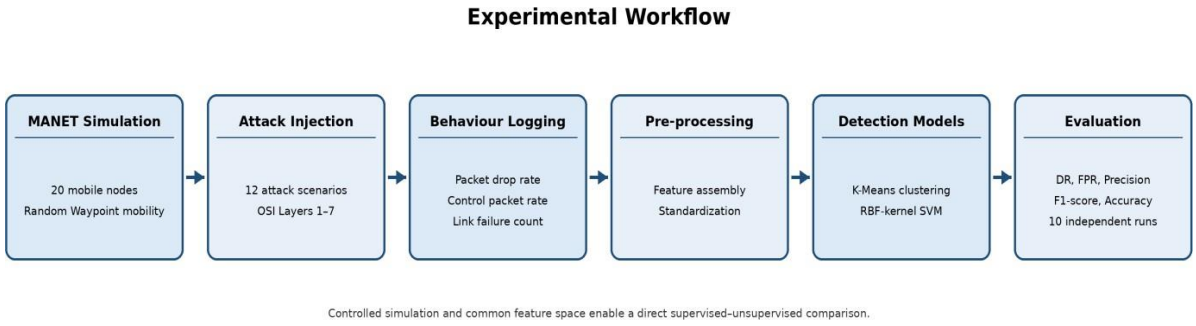


**Figure 1. Experimental workflow for MANET simulation and intrusion-detection evaluation.**

### 3.1 Network Simulation Framework

SimPy, a process-oriented discrete-event simulator for Python, was used to create the

MANET simulation environment. This tool offers various fundamental constructs to model the active components as processes which communicate with each other using shared resources and events [21]. The generator-based execution model of SimPy efficiently represents concurrent network events, including packet transmission, node mobility, and attack injection [21].

NetworkX, a Python library is used in order to construct, modify and examine intricate networks, is employed for the dynamic management of the network topology that enables the real-time reconstruction of graph per changing positions of nodes and constraints of transmission range [22].

The MANET under simulation consists of 20 mobile nodes geographically distributed over an area of 100 x 100 units in a square layout. For each node, the transmission range is the same, equal to 25 units and it is able to communicate wirelessly with all nodes around it since their Euclidean distance is within the threshold distance. Mobility of nodes is modeled with the Random Waypoint. As per this model, each node selects its movement path randomly and only independently. The process is that first the node selects a destination randomly inside the area of operation; then it moves to that destination with a speed randomly selected; finally, it pauses for a fixed time duration before the next move. And then the whole process is repeated indefinitely [23]. Random waypoint is a model that introduces the key features of random movement of nodes, at the same time it still allows the analysis based on simulation due to its simplicity.

The network graph is reconstructed by the Network Manager module at any point when the simulation is paused. The graph is reconstructed from current node locations, where nodes are joined by an edge if and only if the distance between them in the Euclidean metric is not more than the range of transmission. This continuous remodeling of the topology has the effect of the IDS working with the real-time accurate network condition that corresponds to the time-varying connectivity patterns being typical of the Mobile Ad Hoc Networks (MANETs).

Table 1 shows the principal simulation and model parameters used.

**Table 1. Principal simulation and model parameters.**

| Parameter | Setting |
|---|---|
| Mobile nodes | 20 |
| Simulation area | 100 × 100 units |
| Transmission range | 25 units |
| Mobility model | Random Waypoint |
| Attack scenarios | 12 |

| | |
|---|---|
| Observation features | Packet drop rate, control packet rate, link failure count |
| Independent runs per attack | 10 |
| K-Means clusters | k = 2 |
| SVM kernel | Radial basis function |
| SVM regularization | C = 10.0 |
| Implementation | Python; SimPy; NetworkX; NumPy; Pandas; scikit-learn |

The complete simulation framework was implemented in Python using SimPy for discrete-event simulation, NetworkX for dynamic graph management, NumPy and Pandas for data processing, and Scikit-learn for implementing the K-Means clustering and SVM classifiers.

### 3.2 Attack Modeling and Scenario Design

The study was designed with twelve unique attack scenarios targeting OSI layers 1, 3 to 7, effectively addressing the various threat layers in a MANET environment. To simulate the attacks, certain nodes in the network were converted into malicious ones that behaved differently per the type of attack. Table 2 summarizes the implemented attack scenarios and their corresponding OSI layers.

**Table 2: Multi-Layer Attack Scenarios Implemented in the Simulation Framework [4],[9],[10]**

| Attack Type | OSI Layer | Behavioral Description |
|---|---|---|
| Jamming | L1 | Wireless channel disruption causing link failures |
| Blackhole | L3 | Attracts and drops all routed packets |
| Wormhole | L3 | Creates unauthorized tunnels between distant nodes |
| Flooding | L3 | Excessive control traffic generation |
| Sybil | L3 | Multiple forged identities for single node |
| Sinkhole | L3 | Lures traffic toward compromised node |
| Greyhole | L3 | Selective probabilistic packet dropping |
| Hijacking | L3 | Unauthorized route interception |
| SYN Flood | L4 | TCP connection state exhaustion |
| Session Hijack | L5 | Unauthorized session takeover |
| Decrypt Bypass | L6 | Encryption mechanism circumvention |
| APP DDoS | L7 | Application-layer service flooding |

The network layer attacks (Layer 3) are Blackhole assaults where malicious nodes present false routing information to attract traffic and drop intercepted packets, Greyhole attacks based on selective packet dropping which is a method to evade detection, Sinkhole attacks that attract network traffic to compromised nodes and Wormhole attacks where unauthorized low-latency tunnels between distant network regions are created to disrupt routing topology [4], [9], [24]. The Jamming attack is a physical layer disruption that targets, making wireless channel unavailable. Flooding attack generates a lot of control traffic to deplete network

resources [4]. Transport layer attack includes SYN Flood (Layer 4) which targets TCP connection state tables. Session Hijack (Layer 5) finds session management weaknesses to exploit. Decrypt Bypass (Layer 6) tries to get around encryption schemes. Lastly, Application-layer DDoS (Layer 7) uses a flood of malicious requests to a particular service [10].

### 3.3 Feature Extraction and Behavioral Profiling

Intrusion detection involves identifying behavioral features that separate normal network operation from malicious ones. After studying the unique behavioral patterns shown by the simulated attack scenarios, three main features were chosen for their low computational cost and high separation power: packet drop rate, control packet rate and link failure count. Packet drop rate represents the proportion of packets that fail to reach their destinations and is particularly sensitive to Blackhole, Greyhole, and Sinkhole attacks. Control packet rate determines the number of routing control messages about data traffic and thus it represents the abnormal signaling behavior caused by flooding and wormhole attacks. Link failure count keeps a record of the number of wireless link failures and therefore, it is quite sensitive to jamming and mobility-related artifacts.

These indicators are consistent with the use of compact behavioural evidence for packet-dropping detection and anomaly analysis [7,28].

Packet drop $\text{rate}_i = D_i / T_i$ Control packet $\text{rate}_i = C_i / \Delta t$

Link failure $\text{count}_i$ = number of failed links involving node i during $\Delta t$

Here, $D_i$ is the number of packets dropped by node i, $T_i$ is the number of packets presented to that node for forwarding, $C_i$ is the number of control packets generated, and $\Delta t$ is the observation interval.

The feature extraction proceeds at predetermined logging intervals maintained throughout the simulation. Each node will have a behavioral profile constituted by the three extracted features. The accumulated logs are then assembled into a Data Frame, a structured format ready for ML analysis. The supervised Support Vector Machine (SVM) requires ground-truth class labels indicating the attacker status of each instance during the training phase. In contrast, the unsupervised K-Means algorithm performs clustering without prior knowledge of which instances correspond to attacker or normal nodes.

### 3.4 K-Means Clustering-Based Intrusion Detection

The unsupervised intrusion detection method uses the K-Means clustering approach to split

network nodes into two groups based on their behavioral feature profiles. K-Means was chosen because it is computationally efficient. In fact, the time taken by K-Means depends linearly on the size of the dataset. Another advantage of the algorithm is it doesn't require labeled training data [26]. The algorithm performs the usual iterative cycle of center initialization, sample assignment to the nearest centers and center recalculation until the convergence criteria are met.

$$J = \sum_{i=1}^{k} \sum_{x \in C_i} \| x - \mu_i \|^2$$

where:

- $J$is the clustering objective function,
- $k$is the number of clusters,
- $C_i$denotes the $i^{th}$cluster,
- $x$is a data point, and
- $\mu_i$represents the centroid of cluster $C_i$.

After clustering, the cluster assignment labels need to be interpreted for identifying which cluster depicts malicious behavior. The suggested method arranges the feature values averages of each cluster and selects the cluster having the higher average anomaly-related feature values (high drop rate, increased control traffic and excessive link failures) as the anomalous cluster. The nodes that belong to this cluster are considered as potential intruders. This post-clustering labeling method allows a completely unsupervised mode of operation without having to know attack patterns or malicious node identities beforehand.

### 3.5 Support Vector Machine-Based Intrusion Detection

The proposed Supervised Intrusion Detection model employs a Radial Basis Function (RBF) kernel Support Vector Machine classifier, set with a regularization parameter C = 10.0. SVMs were chosen based on a few reasons: through research they do be very successful for the classification of high-dimensional data; they have robust generalization capabilities by maximizing the margin of the hyperplane in the feature space; and they have always been used as one of the best methods for detecting network intrusions [27]. The RBF kernel makes it possible to have nonlinear decision boundaries that are useful largely for discovering

the complicated connections between the behavioral features and the diverse attack patterns. The decision function is:

$$f(x) = \text{sign}\left(\sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b\right)$$

where,

- $\alpha_i$are Lagrange multipliers,
- $y_i$are class labels,
- $K(x_i, x)$is the kernel function,
- $b$is the bias term.

The RBF kernel is:

$K(x_i, x_j) = \exp(-\gamma \parallel x_i - x_j \parallel^2)$

where $\gamma$controls the influence of individual training samples.

The extracted features are standardized using the Standard Scaler technique before model training. This process transforms each feature to have a zero mean and unit variance, thereby eliminating the influence of differences in feature scales. Standardization ensures that no single feature dominates the learning process solely because of its numerical range, allowing all features to contribute equally to model training. In the supervised learning approach, the model is trained using the ground-truth class labels, enabling it to learn the distinguishing characteristics of normal and malicious node behaviour and accurately classify previously unseen instances.

### 3.6 Performance Evaluation Metrics

Detection performance is assessed by some frequently utilized metrics, which are basically derived from the elements of the confusion matrix: True Positives (TP) are attackers who have been correctly identified; True Negatives (TN) are normal nodes that have been accurately classified; False Positives (FP) are normal nodes that have been improperly marked as attackers; and False Negatives (FN) are attackers who are not detected at all. By using these basic numbers, the following evaluation metrics have been determined:

**Detection Rate (Recall):** Detection Rate (DR) or Recall or True Positive Rate shows how many of the real attackers have been successfully identified by the detection system.

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

**False Positive Rate:** False Positive Rate (FPR) tells what percentage of normal nodes have been wrongly classified as attackers.

$$FPR = \frac{FP}{FP + TN}$$

**Precision:** Precision is the fraction of the nodes flagged as malicious ones that truly are the attackers.

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

**F1-Score:** F1-Score is the harmonic mean of precision and recall, which expresses both the aspects of detection equally.

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

**Accuracy:** The ratio of all right answers is called accuracy classifications to all nodes.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

To make the results statistically reliable, each attack situation is re-run through a computer simulation 10 times and the results are obtained from the average values. In addition to attack-specific results, pooled overall scores were used to summarize the models. Because accuracy can be inflated when normal observations outnumber malicious observations, precision, recall, F1-score, and attack-level results were considered together rather than relying on accuracy alone.

## 4. RESULTS AND DISCUSSION

This section reports the attack-specific and overall performance of the K-Means clustering and radial basis function (RBF)-kernel Support Vector Machine (SVM) intrusion detection

models. All values represent averages obtained from ten independent simulation runs. The discussion focuses on detection effectiveness, false alarms, attack-wise behaviour, and the extent to which the three selected features—packet drop rate, control packet rate, and link failure count—support multi-layer intrusion detection in Mobile Ad Hoc Networks (MANETs).

### 4.1 K-Means Results

Table 3 summarises the performance of the unsupervised K-Means-based intrusion detection system across the twelve attack scenarios. K-Means achieved a detection rate of 1.000 with no reported false positives for Jamming, Flooding, Sinkhole, and SYN Flood. These attacks generated pronounced changes in at least one of the selected behavioural features, allowing the anomalous cluster to be separated clearly from normal node behaviour.

**Table 3: K-Means IDS Performance Result.**

| Attack | DR | FPR | Precision | F1-score | Accuracy |
|---|---|---|---|---|---|
| Jamming | 1.000 | 0.0000 | 1.0000 | 1.0000 | 1.000 |
| Blackhole | 0.500 | 0.3474 | 0.3188 | 0.3343 | 0.645 |
| Wormhole | 0.400 | 0.4389 | 0.0896 | 0.1459 | 0.545 |
| Flooding | 1.000 | 0.0000 | 1.0000 | 1.0000 | 1.000 |
| Sybil | 0.300 | 0.5368 | 0.0248 | 0.0458 | 0.455 |
| Sinkhole | 1.000 | 0.0000 | 1.0000 | 1.0000 | 1.000 |
| Greyhole | 0.500 | 0.4842 | 0.0591 | 0.1045 | 0.515 |
| Hijacking | 0.300 | 0.3730 | 0.0792 | 0.1175 | 0.610 |
| SYN Flood | 1.000 | 0.0000 | 1.0000 | 1.0000 | 1.000 |
| Session Hijack | 0.700 | 0.4158 | 0.1674 | 0.2205 | 0.590 |
| Decrypt Bypass | 0.600 | 0.5684 | 0.0502 | 0.0925 | 0.440 |
| App DDoS | 0.400 | 0.4684 | 0.0361 | 0.0660 | 0.525 |

Performance decreased substantially for attacks whose behavioural profiles overlapped with normal activity. The lowest F1-scores were obtained for Sybil (0.0458), Application DDoS (0.0660), Decrypt Bypass (0.0925), Greyhole (0.1045), Hijacking (0.1175), and Wormhole (0.1459). Decrypt Bypass produced the highest false-positive rate (0.5684), followed by Sybil (0.5368) and Greyhole (0.4842). These results indicate that a two-cluster partition can identify attacks that create strong aggregate anomalies, but it is less reliable when malicious behaviour is subtle, intermittent, or not directly represented by the selected features.

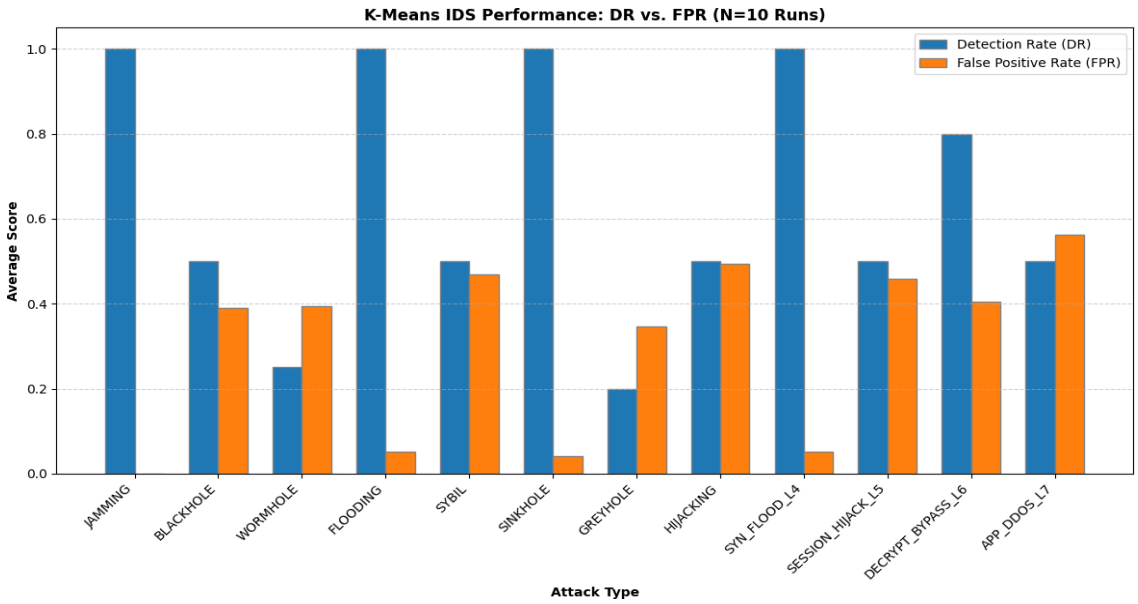


**Figure 2. K-Means IDS Performance: DR vs. FPR**

Figure 2 shows the Detection Rate (DR) and False Positive Rate (FPR) of an intrusion detection system based on K-Means for different types of attack scenarios averaged over ten simulation runs. The figures indicate that attacks such as Jamming, Flooding, Sinkhole, and SYN_FLOOD_L4 achieve very high detection rates, with detection ratios approaching 1.0. In contrast, Greyhole and Wormhole attacks exhibit comparatively lower detection rates. This difference can be attributed to the behavioural characteristics of these attacks, as they closely resemble normal network activities, making them more difficult to distinguish and detect. Application-layer DDoS produced the highest false positive rate, indicating that legitimate traffic was more frequently misclassified as malicious because of behavioural similarities between normal and attack traffic. In general, K-Means algorithm was successful in detecting attacks that caused very noticeable changes in network behaviour e.g. packet drops, control packet activity and link failures; Still, it was relatively less effective for stealthy attacks that are almost indistinguishable from normal MANET operations.

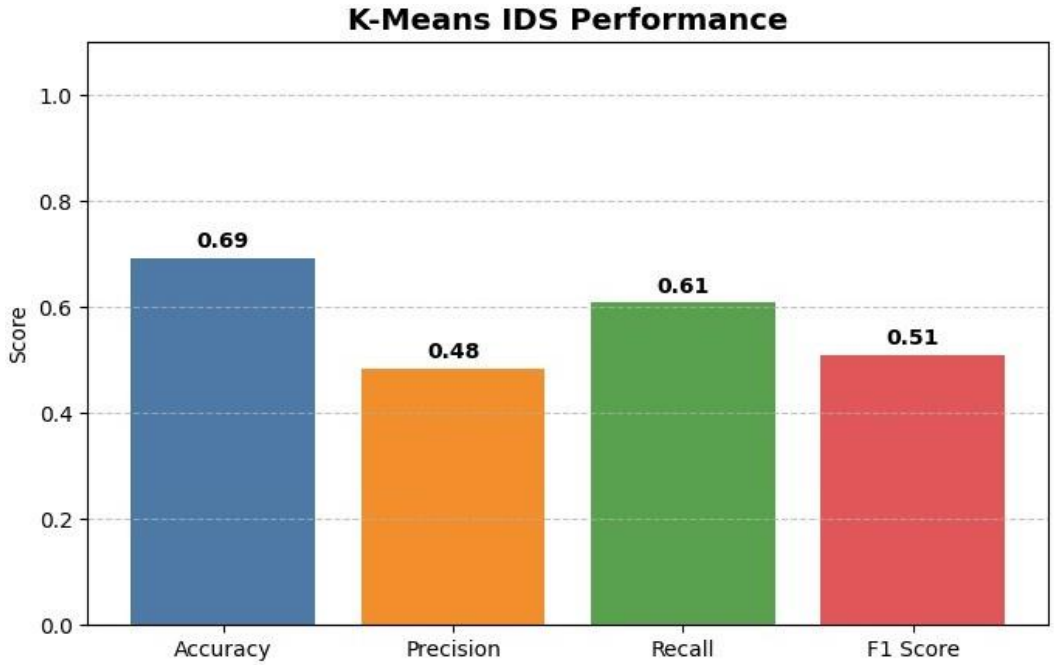


**Figure 3. K-Means IDS Performance**

Overall, the effectiveness of the K-Means-based Intrusion Detection System (IDS) was demonstrated in Figure 3 using four evaluation metrics: Accuracy Precision Recall and F1-Score. The model's classification accuracy was 0.69, which means that nearly 69% of the samples were categorized correctly. Model's skill, as measured by recall, to discover malicious activities has a moderate degree of 0.61 and it means, that a significant proportion (but not all) of actual attacks were captured by the system. The precision score of 0.48 suggests that a major portion of predictions that were made as malicious are not genuine (i.e. many false alarm alerts have been generated). The obtained F1-score of 0.51 indicates a moderate balance between precision and recall. These types of scores suggest that the performance of the algorithm K-Means is reasonably well with the major disadvantage of the inability to effectively separate various or hidden attack patterns by using unsupervised clustering alone.

### 4.2 SVM Results

**5. Table 4 : SVM IDS Performance Result.**

| Attack | DR | FPR | Precision | F1-score | Accuracy |
|---|---|---|---|---|---|
| Jamming | 1.000 | 0.0000 | 1.0000 | 1.0000 | 1.000 |
| Blackhole | 1.000 | 0.0000 | 1.0000 | 1.0000 | 1.000 |
| Wormhole | 0.050 | 0.0000 | 0.1000 | 0.0667 | 0.905 |
| Flooding | 0.900 | 0.0000 | 0.9000 | 0.9000 | 0.995 |
| Sybil | 0.000 | 0.0000 | 0.0000 | 0.0000 | 0.950 |
| Sinkhole | 1.000 | 0.0000 | 1.0000 | 1.0000 | 1.000 |

| | | | | | |
|---|---|---|---|---|---|
| Greyhole | 0.900 | 0.0000 | 0.9000 | 0.9000 | 0.995 |
| Hijacking | 0.100 | 0.0000 | 0.1000 | 0.1000 | 0.955 |
| SYN Flood | 1.000 | 0.0000 | 1.0000 | 1.0000 | 1.000 |
| Session Hijack | 0.700 | 0.0000 | 0.7000 | 0.7000 | 0.985 |
| Decrypt Bypass | 1.000 | 0.0000 | 1.0000 | 1.0000 | 1.000 |
| App DDoS | 0.700 | 0.0000 | 0.7000 | 0.7000 | 0.985 |

Table 4 shows the results of the SVM-based Intrusion Detection System for detecting different attacks by using Detection Rate (DR), False Detection Rate (FPR) Precision F1-Score and Accuracy metrics. The findings reveal that the attacks of Jamming Blackhole Sinkhole, Syn-Flood-L4 and Decrypt-Bypass-L6 had the highest scores for performance, with perfect detection rates, precisions and F1-scores equaling 1.0, which proved the ability of SVM to detect attacks that create major behavioural changes in the network. However, Flooding and Greyhole attacks also performed well and had F1-scores of 0.90, while Session-Hijack-L5 and App-DDOS-L7 had only moderate detection capacities with F1-scores of 0.70. Wormhole, Hijacking and Sybil attacks, showed poor detection results, with the Sybil attack having an F1-score of 0.0 as it is almost indistinguishable from normal node behaviour. In general, research finds that SVM properly identifies attacks with different behaviours but has difficulties in identifying very stealthy and topology-based attacks in MANETs.

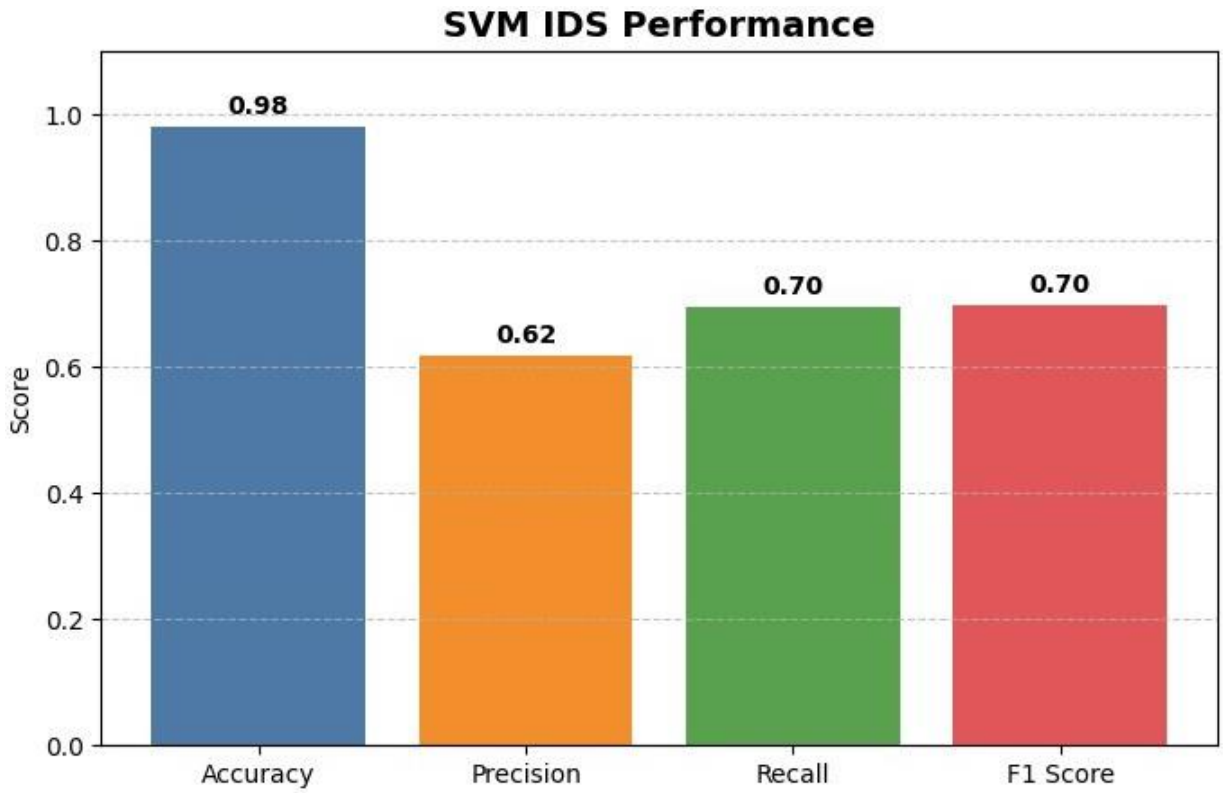


**Figure 4. SVM IDS Performance: DR vs. FPR.**

Figure 4 offers results of a complete performance evaluation of the Support Vector Machine (SVM)-based IDS using four metrics: Accuracy Precision Recall F1-score. The high accuracy of 0.98 means that the machine learning model has identified almost 98% of the attacks correctly. The moderate precision (0.62) of the model suggests that it tried to reduce the percentage of false positives by a moderate amount. A good balance is achieved if it has detected most of the damages without making any mistakes, which is the case here with a recall of 0.70. The F1-score (0.70) is a tradeoff between precision and recall so the overall system detection ability has a very promising improvement. The SVM supervised learning approach has turned out better. Strategy can successfully recognize very complicated behavioural patterns and have a dependable way of detecting intrusions due to multi-layer attacks in MANETs.

### 5.1 Comparative Results

Comparative Analysis of K-Means and SVM IDS is as follows:

**Table 5. Comparative Performance of K-Means and SVM IDS.**

| Metric | K-Means | SVM |
|---|---|---|
| Detection rate (recall) | 0.61 | 0.70 |
| False-positive rate | 0.30 | 0.00 |
| Precision | 0.48 | 0.62 |
| F1-score | 0.51 | 0.70 |
| Accuracy | 0.69 | 0.98 |

Table 5 presents the overall performance comparison between the K-Means and SVM-based intrusion detection systems. The comparison indicates that SVM consistently achieved higher Detection Rate, Precision, F1-Score, and Accuracy while maintaining a False Positive Rate close to zero. In contrast, K-Means showed comparatively lower detection performance and higher false positive rates because it performs unsupervised clustering without using labelled training data. Overall, SVM demonstrated superior effectiveness for identifying malicious nodes in the simulated MANET environment.

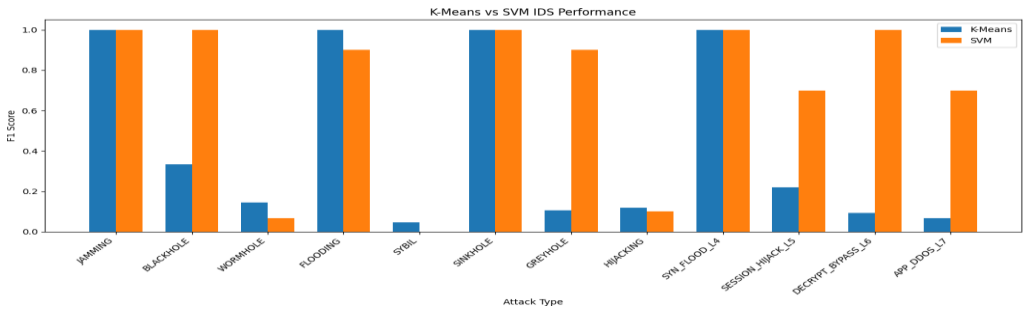


**Figure 5. K-Means vs SVM IDS Performance.**

Figure 5 visually compares the overall performance of K-Means and SVM across the evaluated metrics. The graph confirms that SVM consistently outperformed K-Means in terms of Detection Rate, Precision, F1-Score, and Accuracy while maintaining a significantly lower False Positive Rate. These results demonstrate that supervised learning is more effective than unsupervised clustering for intrusion detection in MANET environments, particularly for identifying complex multi-layer attacks. It illustrates the results of comparing the performances of the K-Means and Support Vector Machine (SVM) based Intrusion Detection Systems (IDS) in different MANET attack situations as measured by the F1-score. Overall, the findings showed that the SVM was more effective than the K-Means in most of the attack categories, which indicates that SVM has more accurate classification and better learning of the attack behavior. Despite this, both methods performed equally well for Jamming, Sinkhole and SYN Flood-L4 with an F1-score of 1.0. SVM also managed to detect Blackhole Greyhole Session Hijack-L5, Decrypt Bypass-L6 and APP DDoS-L7 attacks. For these attacks, SVM was superior to K-Means, whose score was lower than that of SVM. Both methods were not good at recognizing a Wormhole Attack or a Hijacking Attack since the behavior of these is similar like the behavior of the normal traffic flow and so they are hard to be distinguished. But the outcomes reveal that training with SVM leads to better detection of intrusion when comparing to clustering with K-Means when uncovering a multi-layer attack in MANETs that are completely unsupervised.

Overall, the experimental evaluation demonstrates that the supervised SVM classifier consistently outperformed the unsupervised K-Means algorithm across most attack scenarios. While both methods effectively detected attacks producing significant behavioural changes, SVM achieved higher overall accuracy, precision, and F1-score. The results also highlight the continued challenge of detecting attacks such as Wormhole and Greyhole, indicating the need for richer behavioural features and hybrid intrusion detection techniques in future MANET security research.

## 5.2 DISCUSSION

The comparison demonstrates the complementary strengths of supervised and unsupervised intrusion detection. K-Means does not require labelled training data and can therefore provide a useful initial anomaly-screening mechanism. Its two-centroid representation, however, assumes that normal and malicious observations form sufficiently distinct groups. When several attack types share feature ranges with normal traffic, this assumption produces mixed clusters, reduced precision, and elevated false-positive rates. In contrast, the RBF-kernel

SVM uses labelled examples to construct a nonlinear decision boundary and can model more complex combinations of the three behavioural features.

The attack-wise results further clarify the role of feature selection. Jamming directly increases link failures; Flooding and SYN Flood produce unusually high signalling or connection activity; and Blackhole and Sinkhole generate pronounced packet-forwarding anomalies. These behaviours are reflected reasonably well by the selected features. Wormhole, Sybil, and Hijacking attacks can alter topology, route selection, or identity without consistently changing the three aggregate indicators. Their poor detection therefore reflects a feature-representation limitation rather than only an algorithmic limitation. Similar challenges have motivated two-stage MANET intrusion detection architectures [29], class-imbalance-aware frameworks [30], and protocol-specific Blackhole defences based on dynamic thresholds or digital signatures [31,32].

The reported SVM performance is promising, but the controlled synthetic environment limits the strength of the generalisation claim. A robust evaluation should separate training and testing at the simulation-run level, report class distributions and confusion matrices, and provide confidence intervals or standard deviations across runs. These additions would help determine whether the observed gains remain stable under different node densities, mobility conditions, traffic loads, and attacker proportions.

### 5.2.1. Limitations

This study has five principal limitations. First, the simulation is a behavioural abstraction and does not reproduce every radio, medium-access-control, transport, and application-layer operation of a packet-level simulator or physical testbed. Second, the experiment uses only 20 nodes and one mobility model, which limits conclusions regarding scalability and mobility diversity. Third, the compact three-feature representation excludes routing-path changes, neighbour trust, identity consistency, temporal sequences, signal strength, and packet-level content. Fourth, the findings are based on synthetic attack scenarios and require validation using independent datasets, emulation platforms, or real testbed traffic. Fifth, the high SVM accuracy and reported zero false-positive rate require confirmation through explicitly separated training and testing runs, repeated cross-validation, class-distribution reporting, and statistical uncertainty estimates.

### 5.2.2. Practical Implications and Future Work

A practical MANET intrusion detection system could use the three lightweight behavioural

features as a first-stage screening mechanism and activate a richer second-stage analysis only for suspicious nodes. This arrangement would preserve low computational overhead while improving the detection of topology- and identity-oriented attacks. Future work should incorporate temporal windows, route-change frequency, hop-count variation, neighbour-table inconsistency, trust scores, signal-quality measures, and cross-layer dependencies.

Future evaluation should vary network size, attacker proportion, node speed, pause time, transmission range, and traffic load; report confidence intervals and significance tests; and measure detection latency, memory use, computational cost, communication overhead, and energy consumption. A hybrid architecture that combines unsupervised novelty detection with supervised attack classification may offer a better balance between discovering unknown threats and accurately identifying known attacks. Privacy-preserving distributed learning [33] and lightweight deep models [34] should be investigated only when their computational and communication requirements remain compatible with resource-constrained nodes.

## 6. CONCLUSION

This paper presented a comparative evaluation of K-Means clustering and Support Vector Machine (SVM) techniques for intrusion detection in Mobile Ad Hoc Networks (MANETs). A Python-based simulation framework incorporating twelve attack scenarios across multiple OSI layers, specifically Layer 1 and Layers 3–7, was developed to evaluate both approaches under identical experimental conditions. The models were assessed using three lightweight behavioural features: packet drop rate, control packet rate, and link failure count.

The experimental results showed that the supervised SVM classifier achieved better overall performance than the unsupervised K-Means algorithm in terms of detection rate, precision, F1-score, accuracy, and false-positive rate. SVM achieved an overall accuracy of 0.98 and an F1-score of 0.70, compared with 0.69 and 0.51, respectively, for K-Means. Both methods effectively detected attacks that produced clear behavioural deviations, including Jamming, Sinkhole, and SYN Flood. However, attacks such as Wormhole, Sybil, and Hijacking remained difficult to detect because their effects were not adequately represented by the selected behavioural features. Although K-Means performed comparatively better for a small number of individual attack scenarios, SVM provided stronger and more reliable overall discrimination when labelled training data were available.

The findings demonstrate that packet drop rate, control packet rate, and link failure count provide a computationally lightweight basis for detecting several MANET attacks. Nevertheless, these features alone are insufficient for accurately identifying identity-based,

topology-oriented, and route-manipulation attacks. Future research should therefore incorporate routing-path variations, neighbour-table inconsistencies, node identity behaviour, trust measures, temporal communication patterns, signal characteristics, and cross-layer information. Hybrid intrusion detection frameworks combining unsupervised anomaly detection with supervised attack classification may further improve the detection of both known and previously unseen attacks while maintaining the computational efficiency required in resource-constrained MANET environments.

## REFERENCES


1. R. Vatambeti, S. V. Mantena, K. Kasula, S. Chennupalli and M. V. Gopalachari, "Black Hole Attack Detection Using Dolphin Echo-Location-Based Machine Learning Model in MANET Environment", *Computers & Electrical Engineering*, vol. 114, Art. no. 109094, 2024, doi: 10.1016/j.compeleceng.2024.109094.
2. A. H. Mohsin, "Optimize Routing Protocol Overheads in MANETs: Challenges and Solutions: A Review Paper", *Wireless Personal Communications*, vol. 126, no. 4, pp. 2871–2910, 2022, doi: 10.1007/s11277-022-09843-3.
3. K. S. Nirmala Bai and M. V. Subramanyam, "Integrated intrusion detection design with discretion of leading agent using machine learning for efficient MANET system", Scientific Reports, vol. 15, Article 30849, 2025. DOI: 10.1038/s41598-025-89221-8
4. S. M. Hassan, M. M. Mohamad and F. B. Muchtar, "Advanced Intrusion Detection in MANETs: A Survey of Machine Learning and Optimization Techniques for Mitigating Black/Gray Hole Attacks", *IEEE Access*, vol. 12, pp. 150046–150090, 2024, doi: 10.1109/ACCESS.2024.3378891.
5. S. A. Alghamdi, "Novel Trust-Aware Intrusion Detection and Prevention System for 5G MANET–Cloud", *International Journal of Information Security*, vol. 21, no. 3, pp. 469–488, Jun. 2022, doi: 10.1007/s10207-020-00531-6.
6. M. Sahaya Sheela, A. Gnana Soundari, A. Mudigonda, C. Kalpana, et al., "Adaptive Marine Predator Optimization Algorithm (AOMA)-Deep Supervised Learning Classification (DSLC) Based IDS Framework for MANET Security", *Intelligent and Converged Networks*, vol. 5, no. 1, pp. 1–18, Mar. 2024, doi: 10.23919/ICN.2024.0001.
7. S. Djahel, F. Naït-Abdesselam and Z. Zhang, "Mitigating Packet Dropping Problem in Mobile Ad Hoc Networks: Proposals and Challenges", *IEEE Communications Surveys & Tutorials*, vol. 13, no. 4, pp. 658–672, 2011, doi: 10.1109/SURV.2011.072210.00026.
8. H. S. Jassim, R. K. Al Majizi, Z. T. Sharef and S. A. Jawdat, "Black-Hole and DoS Attack

Detection Analysis and Isolation Mechanism for Black-Hole in MANET", *Open Access Research Journal of Engineering and Technology*, vol. **7**, no. 1, pp. 1–13, 2024, doi: 10.53022/oarjet.2024.7.1.0026.

9. Z. B. Ibrahim and M. F. Ghanim, "A Review of AI-Based Approaches against Wormhole and Blackhole Attacks in AODV Protocol", *International Journal of Advanced Natural Sciences and Engineering Researches*, vol. 8, no. 5, pp. 60–75, 2024.
10. I. H. Sarker**,** *AI-Driven Cybersecurity and Threat Intelligence: Cyber Automation, Intelligent Decision-Making and Explainability*. Cham, Switzerland: Springer Nature Switzerland, 2024. doi: 10.1007/978-3-031-54497-2.
11. A. Aldweesh, A. Derhab and A. Z. Emam**,** "Deep Learning Approaches for Anomaly-Based Intrusion Detection Systems: A Survey, Taxonomy and Open Issues", *Knowledge-Based Systems*, vol. 189, Art. no. 105124, 2020, doi: 10.1016/j.knosys.2019.105124.
12. S. Hemalatha, K. V. S. V. Trinadh Reddy, Ramaswamy T., R. V. V. Krishna, P. Supriya and S.
13. N. Ananthi, "Enhancing MANET Security Using AI-Driven Intrusion Detection Systems",
14. *Journal of Communications*, vol. 20, no. 4, pp. 371–383, 2025, doi: 10.12720/jcm.20.4.371-383.
15. S. Rajapaksha, H. Kalutarage, M. O. Al-Kadri, A. Petrovski, G. Madzudzo and M. Cheah, "AI-Based Intrusion Detection Systems for In-Vehicle Networks: A Survey", *ACM Computing Surveys*, vol. 55, no. 11, Art. no. 237, pp. 1–40, 2023, doi: 10.1145/3570954.
16. L. Bai, P. Han, J. Wang and J. Wang, "Throughput Maximization for Multipath Secure Transmission in Wireless Ad-Hoc Networks", *IEEE Transactions on Communications*, vol. 72, no. 11, pp. 6810–6821, Nov. 2024, doi: 10.1109/TCOMM.2024.3409539.
17. U. Ahmed, M. Nazir, A. Sarwar, T. Ali, E.-H. M. Aggoune, T. Shahzad and M. A. Khan, "Signature-Based Intrusion Detection Using Machine Learning and Deep Learning Approaches Empowered with Fuzzy Clustering", *Scientific Reports*, vol. 15, Art. no. 1726, 2025, doi: 10.1038/s41598-025-85866-7.
18. Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou and C. Wang, "Machine Learning and Deep Learning Methods for Cybersecurity", *IEEE Access*, vol. 6, pp. 35365–35381, 2018, doi: 10.1109/ACCESS.2018.2836950.
19. M. Yazdanypoor, S. Cirillo and G. Solimando, "Developing a Hybrid Detection Approach to Mitigating Black Hole and Gray Hole Attacks in Mobile Ad Hoc

Networks", *Applied Sciences*, vol. 14, no. 17, Art. no. 7982, 2024, doi: 10.3390/app14177982.

20. S. Faizal Mukthar Hussain and S. M. H. Sithi Shameem Fathima, "Federated Learning-Assisted Coati Deep Learning-Based Model for Intrusion Detection in MANET", *International Journal of Computational Intelligence Systems*, vol. 17, Art. no. 285, 2024, doi: 10.1007/s44196-024-00590-w.
21. J. Zhu and X. Liu, "An Integrated Intrusion Detection Framework Based on Subspace Clustering and Ensemble Learning", *Computers & Electrical Engineering*, vol. 115, Art. no. 109113, 2024, doi: 10.1016/j.compeleceng.2024.109113.
22. A. John, I. F. Bin Isnin, S. H. H. Madni and M. Faheem, "Intrusion Detection in Cluster-Based Wireless Sensor Networks: Current Issues, Opportunities and Future Research Directions", *IET Wireless Sensor Systems*, vol. 14, no. 6, pp. 293–332, 2024, doi: 10.1049/wss2.12100.
23. D. Zinoviev, "Discrete Event Simulation: It's Easy with SimPy!", *arXiv preprint*arXiv:2405.01562, 2024, doi: 10.48550/arXiv.2405.01562.
24. A. A. Hagberg, D. A. Schult and P. J. Swart, "Exploring Network Structure, Dynamics and Function Using NetworkX", in *Proc. 7th Python in Science Conference (SciPy 2008)*, Pasadena, CA, USA, 2008, pp. 11–15.
25. M. K. Pandey and S. Kandari, "Random WayPoint Mobility Model Based Performance Estimation of MANET in Terms of Average End-to-End Delay, Jitter and Throughput for CBR Application", *International Journal of Computer Applications*, vol. 106, no. 3, pp. 1–7, Nov. 2014, doi: 10.5120/18497-9562.
26. R. Khalladi, M. Rebbah and O. Smail, "A New Efficient Approach for Detecting Single and Multiple Black Hole Attacks", *The Journal of Supercomputing*, vol. 77, no. **7**, pp. 7718–7736, 2021, doi: 10.1007/s11227-020-03596-1.
27. C. Li, A. He, G. Liu, Y. Wen, A. T. Chronopoulos and A. Giannakos, "RFL-APIA: A Comprehensive Framework for Mitigating Poisoning Attacks and Promoting Model Aggregation in IIoT Federated Learning", *IEEE Transactions on Industrial Informatics*, vol. 20, no. 11, pp. 12935–12944, Nov. 2024, doi: 10.1109/TII.2024.3431020.
28. J. MacQueen, "Some Methods for Classification and Analysis of Multivariate Observations", in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, Berkeley, CA, USA, 1967, pp. 281–297.
29. C. Cortes and V. Vapnik, "Support-Vector Networks", *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995, doi: 10.1007/BF00994018.

30. S. S. Bagui, G. C. S. De Carvalho, A. Mishra, D. Mink, S. C. Bagui and S. Eager, "Detecting Cyber Threats in UWF-ZeekDataFall22 Using K-Means Clustering in the Big Data Environment", *Future Internet*, vol. 17, no. 6, Art. no. 267, 2025, doi: 10.3390/fi17060267.
31. N. Rajan, M. T. R. Raghu, M. N. Shanmukha Swamy and S. N. Manjunath, "Two-Pronged Intrusion Detection System for MANET", in *Proc. 2024 International Conference on Emerging Technologies in Computer Science for Interdisciplinary Applications (ICETCS)*, Bengaluru, India, 2024, pp. 1–6**,** doi: 10.1109/ICETCS61085.2024.10733429.
32. W. Zhou, C. Xia, T. Wang, X. Liang, W. Lin, X. Li and S. Zhang, "HIDIM: A Novel Framework of Network Intrusion Detection for Hierarchical Dependency and Class Imbalance", *Computers & Security*, vol. 148, Art. no. 104155, 2025, doi: 10.1016/j.cose.2024.104155.
33. V. Mankotia, R. K. Sunkaria and S. Gurung, "DT-AODV: A Dynamic Threshold Protocol Against Black-Hole Attack in MANET", *Sādhanā*, vol. 48, no4., Art. no. 190, 2023, doi: 10.1007/s12046-023-02227-8.
34. M. I. Talukdar, R. Hassan, M. S. Hossen, K. Ahmad, F. Qamar and A. S. Ahmed**,** "Performance Improvements of AODV by Black Hole Attack Detection Using IDS and Digital Signature", *Wireless Communications and Mobile Computing*, vol. 2021, Art. no. 6693316**,** 13 pages, 2021, doi: 10.1155/2021/6693316.
35. S. Mohammadi, A. Balador, S. Sinaei and F. Flammini, "Balancing Privacy and Performance in Federated Learning: A Systematic Literature Review on Methods and Metrics", *Journal of Parallel and Distributed Computing*, vol. 192, Art. no. 104918, 2024, doi: 10.1016/j.jpdc.2024.104918.
36. Z. A. Abbood, D. C. Atilla and C. Aydin, "Intrusion Detection System Through Deep Learning in Routing MANET Networks", *Intelligent Automation & Soft Computing*, vol. 37, no. 1, pp. 269–281, 2023, doi: 10.32604/iasc.2023.035276.