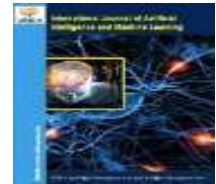




International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: <https://www.svedbergopen.com/>



Research Paper

Open Access

Adaptive Trust-Weighted Federated Threat Learning for Secure and Privacy-Preserving Intrusion Detection in Mobile Ad Hoc Networks

Mr. K. Amuthan¹, Dr. S. K. Mahendran²

¹ Assistant Professor, Department of Computer Technology, Dr. SNS Rajalakshmi College of Arts and Science.
Email: amuthan.k.ct@drsnsrscas.ac.in

² Assistant Professor, Department of Computer Science, Government Arts College (Autonomous), Racecourse, Coimbatore – 641018.
Email: sk.mahendran@yahoo.co.in

Abstract

Mobile Ad Hoc Networks (MANETs) provide decentralized and infrastructure-less communication for applications such as military communication, disaster management, emergency response, intelligent transportation, healthcare monitoring, and Internet of Things environments. However, their dynamic topology, limited resources, and absence of centralized administration make them vulnerable to attacks such as blackhole, wormhole, Sybil, flooding, grayhole, and denial-of-service attacks. Conventional centralized intrusion detection approaches require the transfer of sensitive traffic data to a central server, resulting in privacy, communication, and scalability challenges. Federated Learning (FL) addresses these limitations by allowing distributed nodes to train local models without sharing raw traffic data. However, conventional federated aggregation methods such as FedAvg generally assign equal importance to participating clients and may therefore be vulnerable to unreliable or malicious model updates.

To address these limitations, this paper proposes an Adaptive Trust-Weighted Federated Threat Learning (ATFTL) framework for privacy-preserving intrusion detection in MANETs. The proposed approach introduces a Dynamic Trust Weight (DTW) mechanism that jointly considers the Node Reliability Index (NRI), Residual Energy Score (RES), Communication Stability Factor (CSF), and Historical Detection Accuracy (HDA) of participating nodes. An adaptive malicious-update filtering mechanism further evaluates local model deviation and reduces the contribution of suspicious updates before federated aggregation. The accepted updates are subsequently combined using trust-weighted federated aggregation to generate an optimized global threat intelligence model. The framework was implemented using the CICIDS2017 dataset with distributed local training using a CNN-LSTM model. The simulation evaluates network sizes ranging from 50 to 250 nodes and compares ATFTL with FedAvg and FedMADE using Average DTW, Malicious Update Detection Rate, False Rejection Rate, Trusted Node Ratio, and Communication Overhead. The final threat-detection simulation reports 99.18% accuracy, 96.02% precision, 98.96% recall, 98.99% F1-score, and 0.998 AUC. The comparative results demonstrate that ATFTL improves trust-aware participant selection, malicious-update detection, and reliable federated aggregation while maintaining comparatively lower communication overhead.

Keywords: Mobile Ad Hoc Networks (MANETs); Federated Learning; Intrusion Detection System; Adaptive Trust; Dynamic Trust Weight; Malicious Update Detection; Trust-Weighted Aggregation; Model Poisoning; CNN-LSTM; CICIDS2017

1. Introduction

Mobile Ad Hoc Networks (MANETs) have emerged as one of the most promising wireless networking paradigms due to their infrastructure-less architecture, self-configuring capability, and dynamic topology [1]. These networks are extensively employed in military communications, disaster recovery, emergency response systems, intelligent transportation, healthcare monitoring, and Internet of Things (IoT) applications, where rapid deployment and decentralized communication are essential [2]. Despite these advantages, the absence of centralized administration and continuously changing network topology make MANETs highly susceptible to numerous security threats, including blackhole, wormhole, Sybil, flooding, grayhole, and denial-of-service (DoS) attacks. Such attacks can significantly reduce packet delivery ratio, increase routing overhead, degrade communication quality, and compromise the overall reliability of the network [3].

Intrusion Detection Systems (IDSs) have traditionally been employed to identify malicious activities in MANETs through signature-based, anomaly-based, and hybrid detection techniques. Recent advances in machine learning (ML) and deep learning (DL) have substantially improved intrusion detection accuracy by automatically learning complex attack patterns from network traffic [4]. However, most existing IDS frameworks adopt centralized learning architectures that require distributed nodes to transmit their raw traffic data to a central server for model training. Such centralized approaches suffer from high communication overhead, privacy leakage, scalability limitations, and vulnerability to single-point failures, making them unsuitable for dynamic and resource-constrained MANET environments [5].

Federated Learning (FL) has recently attracted considerable attention as a privacy-preserving distributed learning paradigm that enables multiple nodes to collaboratively train a global model without exchanging raw data[6]. Instead, each participating client performs local model training using its own data and transmits only model parameters to the aggregation server. Consequently, FL significantly enhances privacy preservation, reduces communication costs, and improves scalability in distributed network environments. Recent studies have demonstrated that federated learning achieves intrusion detection performance comparable to centralized deep learning while effectively protecting sensitive information [7].

Although federated learning addresses several challenges associated with centralized IDSs, the majority of existing FL-based intrusion detection frameworks rely on the conventional Federated Averaging (FedAvg) algorithm, which assigns equal aggregation weights to all participating clients irrespective of their reliability, communication quality, or security status. In highly dynamic MANET environments, participating nodes often exhibit heterogeneous characteristics due to variations in residual energy, mobility, routing stability, and communication reliability. Consequently, compromised or malicious nodes can inject poisoned model updates into the aggregation process, thereby degrading the global model, reducing detection accuracy, and slowing model convergence. Existing robust aggregation methods primarily focus on statistical filtering of model gradients while largely ignoring important network-centric trust indicators that directly influence the quality of collaborative learning [8].

Recent research has further emphasized the importance of designing adaptive aggregation strategies capable of mitigating model-poisoning attacks while preserving the collaborative nature of federated learning[9]. Dynamic client selection, explainable federated learning, and adaptive aggregation have demonstrated promising improvements in robustness; however, these approaches rarely incorporate comprehensive trust evaluation based on network behavior, communication stability, residual energy, and historical intrusion detection performance. Moreover, most existing studies have focused primarily on IoT and edge computing environments, whereas trust-aware federated intrusion detection for highly dynamic MANETs remains relatively unexplored [10].

To address these limitations, this paper proposes an Adaptive Trust-Weighted Federated Threat Learning (ATFTL) framework for privacy-preserving intrusion detection in Mobile Ad Hoc Networks. The proposed framework introduces a novel Adaptive Trust Weight (DTW) mechanism that dynamically evaluates the reliability of each participating node using four complementary trust indicators: Node Reliability Index (NRI), Residual Energy Score (RES), Communication Stability Factor (CSF), and Historical Detection Accuracy (HDA). These indicators are integrated to determine the contribution of each node during federated aggregation. Furthermore, an Adaptive Malicious Update Filtering mechanism evaluates the consistency of local model updates before aggregation and proportionally reduces the influence of suspicious nodes rather than completely excluding them from collaborative learning. Finally, a Trust-Weighted Federated Aggregation strategy generates an optimized global threat intelligence model that improves intrusion detection accuracy, enhances robustness against poisoning attacks, accelerates model convergence, and preserves user privacy in highly dynamic MANET environments.

The major contributions of this paper are summarized as follows:

- A novel Adaptive Trust-Weighted Federated Threat Learning (ATFTL) framework is proposed for decentralized and privacy-preserving intrusion detection in MANETs.
- A Dynamic Trust Weight (DTW) model is introduced by integrating Node Reliability Index, Residual Energy Score, Communication Stability Factor, and Historical Detection Accuracy for adaptive client evaluation.
- An Adaptive Malicious Update Filtering mechanism is developed to mitigate poisoned and inconsistent local model updates before global aggregation.
- A Trust-Weighted Federated Aggregation strategy is proposed to improve convergence speed, model robustness, and intrusion detection accuracy compared with conventional FedAvg.
- The proposed framework generates an optimized Global Threat Intelligence Model suitable for secure collaborative intrusion detection and future adaptive threat response in dynamic MANET environments.

2. Literature Review

Hussain *et al.* (2024) proposed a Federated Learning-Assisted Coati Deep Learning Model (FL-CDL) for intrusion detection in Mobile Ad Hoc Networks. Their framework integrated federated learning with deep neural networks to preserve user privacy while improving intrusion detection accuracy. The experimental results demonstrated improved classification performance and reduced communication overhead compared with centralized deep learning methods. However, the framework adopted the conventional Federated Averaging (FedAvg) algorithm, which equally aggregates local model updates without considering node reliability or trustworthiness, making it susceptible to malicious client updates. [11]

Khraisat *et al.* (2025) presented a comprehensive study on privacy-preserving federated intrusion detection systems for IoT and wireless networks. Their work emphasized the importance of federated learning in protecting sensitive network data while enabling collaborative intrusion detection across distributed devices. Although the proposed framework effectively reduced privacy leakage, the aggregation process lacked adaptive trust evaluation and remained vulnerable to model poisoning attacks generated by compromised clients. [12]

Belenguer *et al.* (2025) conducted an extensive review of Federated Learning-based Intrusion Detection Systems, analyzing recent advances in federated optimization, privacy preservation, communication efficiency, and aggregation techniques. The authors reported that most existing intrusion detection frameworks continue to rely on FedAvg or simple weighted averaging strategies, which are insufficient for handling heterogeneous and malicious clients in highly dynamic wireless environments. They recommended the development of trust-aware aggregation algorithms for future federated intrusion detection systems. [13]

Buyuktanir *et al.* (2025) surveyed recent developments in federated intrusion detection and cybersecurity applications. Their study highlighted several research challenges, including client heterogeneity, communication overhead, adversarial attacks, and robustness against poisoned model updates. The authors concluded that adaptive client trust evaluation remains one of the most significant research gaps in federated learning-based intrusion detection systems and suggested trust-driven aggregation as an important future research direction. [14]

Latif *et al.* (2025) reviewed advanced federated learning techniques for intrusion detection by integrating explainable artificial intelligence, neural networks, and privacy-preserving optimization methods. Their survey demonstrated that explainability improves the transparency of intrusion detection decisions; however, the majority of existing approaches evaluate only model gradients during aggregation while ignoring important network characteristics such as communication stability, node reliability, and residual energy. [15]

Zhang *et al.* (2025) presented a systematic survey of Federated Learning for Intrusion Detection Systems, covering data preprocessing, client selection, federated optimization, security mechanisms, and aggregation algorithms. Their analysis revealed that current federated intrusion detection systems experience reduced performance under non-IID data distributions and malicious client participation. The authors emphasized the necessity of designing intelligent trust-aware aggregation mechanisms capable of dynamically adapting to heterogeneous network environments. [16]

Baich *et al.* (2025) proposed a Dynamic Ensemble Aggregation strategy for federated intrusion detection in IoT environments. Their aggregation mechanism combined multiple local models according to prediction performance to improve classification accuracy and robustness. Although the proposed ensemble aggregation outperformed conventional FedAvg, it primarily relied on prediction confidence and did not evaluate client trustworthiness using communication reliability, routing stability, or historical behavior. [17]

Sun *et al.* (2024) introduced FedMADE, a dynamic aggregation framework for federated intrusion detection under non-IID data distributions. Their method clustered participating devices according to traffic similarity and dynamically adjusted aggregation weights to improve minority attack detection. While FedMADE effectively improved intrusion detection accuracy and communication efficiency, it did not incorporate adaptive trust computation based on network conditions or node reliability before aggregation. [18]

Nguyen *et al.* (2024) proposed FedMSE, a semi-supervised federated intrusion detection framework that combines Shrink Autoencoders with Mean Square Error-based aggregation. The proposed method significantly reduced communication costs while improving anomaly detection performance across heterogeneous IoT devices. Nevertheless, the aggregation algorithm primarily optimized prediction accuracy and lacked mechanisms for evaluating communication stability, energy availability, and historical detection behavior of participating clients. [19]

Kalakoti *et al.* (2025) developed FedXAI, a federated explainable artificial intelligence framework for deep learning-based intrusion detection. Their work successfully combined explainability with privacy-preserving federated learning to improve model transparency and user confidence. Despite these improvements, the framework did not introduce an adaptive trust-based aggregation strategy capable of dynamically reducing the influence of unreliable or malicious participants during collaborative model training. [20]

Table 1. Summary of Recent Federated Learning-Based Intrusion Detection Studies

Author(s) & Year [Ref.]	Methodology	Key Findings	Limitations
-------------------------	-------------	--------------	-------------

Hussain et al. (2024) [11]	Federated Learning-Assisted Coati Deep Learning (FL-CDL) for MANET intrusion detection.	Improved privacy and intrusion detection accuracy.	FedAvg treats all nodes equally and may accept malicious updates.
Khraisat et al. (2025) [12]	Privacy-Preserving Federated Intrusion Detection System.	Improved data privacy and collaborative detection.	Does not use adaptive trust evaluation or malicious-client detection.
Belenguer et al. (2025) [13]	Survey of FL-based intrusion detection systems.	Reviewed FL architectures and aggregation methods.	Trust-aware aggregation for unreliable and malicious nodes is still limited.
Buyuktanir et al. (2025) [14]	Survey of federated intrusion detection and cybersecurity.	Identified heterogeneity, communication cost, and poisoning as major challenges.	Does not provide a specific adaptive trust-based solution.
Latif et al. (2025) [15]	Explainable Federated Learning for intrusion detection.	Improved IDS transparency and interpretability.	Does not consider node reliability, energy, communication stability, and history together.
Zhang et al. (2025) [16]	Survey of FL-based IDS architectures and techniques.	Summarized FL-IDS methods and deployment challenges.	Adaptive trust-based aggregation remains insufficiently addressed.
Baich et al. (2025) [17]	Dynamic Ensemble Aggregation for Federated IDS.	Improved classification using ensemble aggregation.	Mainly uses prediction confidence and ignores broader node trust factors.
Sun et al. (2024) [18]	FedMADE dynamic aggregation framework.	Improved minority attack detection under non-IID data.	Does not consider network behavior and communication stability for trust calculation.
Nguyen et al. (2024) [19]	FedMSE semi-supervised federated IDS.	Reduced communication cost and improved anomaly detection.	Does not consider node history, energy, or communication reliability in aggregation.
Kalakoti et al. (2025) [20]	FedXAI: Federated Explainable AI for intrusion detection.	Improved transparency using explainable AI.	Does not address trust-based client selection or trust-weighted aggregation.

2.1 Research Gap

The literature indicates that existing FL-based IDS approaches primarily focus on privacy preservation, detection accuracy, non-IID learning, explainability, communication efficiency, or dynamic aggregation individually. However, they do not adequately integrate multiple node-level trust indicators with model-update deviation analysis. In particular, factors such as Node Reliability Index (NRI), Residual Energy Score (RES), Communication Stability Factor (CSF), and Historical Detection Accuracy (HDA) are rarely considered jointly when assigning aggregation weights.

This gap motivates the proposed Adaptive Trust-Weighted Federated Threat Learning (ATFTL) framework, which combines these trust indicators into a Dynamic Trust Weight (DTW) and subsequently evaluates local model deviation before performing trust-weighted federated aggregation.

3. Proposed Method Adaptive Trust-Weighted Federated Threat Learning (ATFTL)

The proposed Adaptive Trust-Weighted Federated Threat Learning (ATFTL) algorithm is designed to establish a secure, decentralized, and privacy-preserving threat learning mechanism for Mobile Ad Hoc Networks (MANETs). Initially, a global deep learning model $G(r)$ is initialized at the federated server and broadcast to all participating MANET nodes. Each node independently trains the received model using its local network traffic data, thereby ensuring that sensitive traffic information remains within the local device and eliminating the need for centralized data collection. The local model parameters are updated through gradient descent according to $W_i^{r+1} = W_i^r - \eta \nabla L(W_i)$ where η denotes the learning rate and $L(W_i)$ represents the local loss function. This decentralized learning strategy enables every node to capture local traffic characteristics while preserving user privacy and reducing communication overhead.

After local training, each node evaluates its trustworthiness by computing four important trust indicators: the Node Reliability Index (NRI), Residual Energy Score (RES), Communication Stability Factor (CSF), and Historical Detection Accuracy (HDA). The Node Reliability Index is computed from the packet delivery ratio, link stability, and packet drop ratio as

$$NRI_i = \frac{PDR_i + LS_i + (1 - PDrop_i)}{3}$$

while the Residual Energy Score is estimated using $RES_i = \frac{E_{current}}{E_{initial}}$

Similarly, the Communication Stability Factor is calculated as

$$CSF_i = \frac{1}{1 + RC_i}$$

where RC_i denotes the route change frequency, and the Historical Detection Accuracy is determined using

$$HDA_i = \frac{CorrectDetection_i}{TotalDetection_i}$$

These parameters collectively provide a comprehensive evaluation of the communication reliability, energy availability, network stability, and historical intrusion detection capability of each participating node.

Unlike conventional Federated Averaging (FedAvg), where all participating nodes contribute equally to the global model irrespective of their reliability, the proposed ATFTL introduces a novel Adaptive Trust Weight (DTW) to determine the contribution of each node during federated aggregation. The trust weight is computed by integrating the four trust indicators as

$$DTW_i = \alpha NRI_i + \beta RES_i + \gamma CSF_i + \delta HDA_i$$

where $\alpha + \beta + \gamma + \delta = 1$.

This adaptive weighting mechanism ensures that highly reliable nodes exert greater influence on the global model, whereas unreliable or suspicious nodes automatically receive lower aggregation weights. Consequently, the proposed framework significantly improves the robustness of collaborative learning against malicious or poorly performing nodes.

To further enhance the security of federated learning, ATFTL incorporates an Adaptive Malicious Update Filtering mechanism that evaluates the consistency of each local model update before aggregation. The deviation between the local model and the current global model is computed as $Dev_i = \|W_i^{r+1} - G^r\|_2$

If the deviation exceeds a predefined threshold τ , indicating a potentially poisoned or abnormal model update, the corresponding trust weight is penalized using

$$DTW_i = \lambda DTW_i, \text{ where } 0 < \lambda < 1$$

. Instead of completely discarding suspicious model updates, the proposed strategy proportionally reduces their influence on the aggregation process. This adaptive penalty mechanism maintains learning diversity while effectively minimizing the impact of model poisoning attacks and unreliable participants.

Following trust evaluation, the updated trust values are normalized to obtain aggregation coefficients using $\omega_i = \frac{DTW_i}{\sum DTW_j}$ ensuring that the total contribution of all participating nodes equals one. These normalized trust weights are subsequently employed in the proposed Trust-Weighted Federated Aggregation strategy to generate the updated global model according to $G^{r+1} = \sum \omega_i W_i^{r+1}$

Unlike the conventional FedAvg algorithm, which simply computes the arithmetic mean of local models, the proposed aggregation mechanism assigns larger weights to highly trustworthy nodes and smaller weights to suspicious nodes. This trust-aware aggregation significantly improves model convergence, enhances intrusion detection accuracy, and increases resilience against adversarial model manipulation.

Finally, the aggregated global model is utilized to estimate the Threat Confidence Score (TCS) for each attack category using the Softmax function, $TCS_i = \frac{e^{z_i}}{\sum e^{z_k}}$

where z_i represents the prediction score for the i th attack class. Simultaneously, nodes whose trust weights satisfy $DTW_i \geq \theta$ are included in the Trusted Node Repository, while nodes exhibiting lower trust values continue to be monitored in subsequent communication rounds. The ATFTL algorithm therefore produces three important outputs: the Global Threat Intelligence Model, the Threat Confidence Scores, and the Trusted Node Repository, which collectively serve as the knowledge base for Phase II: AdaptiveRL-Guard. By integrating dynamic trust computation, adaptive malicious update filtering, trust-weighted federated aggregation, and confidence-based threat estimation into a unified framework, the proposed ATFTL provides a robust, privacy-preserving, and intelligent federated learning solution capable of significantly enhancing intrusion detection performance in highly dynamic MANET environments.

Algorithm 1: Adaptive Trust-Weighted Federated Threat Learning (ATFTL)

Require: Local traffic datasets $\{D_i\}_{i \in \mathcal{N}}$, initial local model parameters $\{W_i^{(0)}\}_{i \in \mathcal{N}}$, initial global model $G^{(0)}$, learning rate η , communication rounds R , deviation threshold τ , trust penalty factor λ , trust threshold θ , and weighting coefficients $\alpha, \beta, \gamma, \delta$.

Ensure: Optimized Global Threat Intelligence Model G^* and Trusted Node Repository TR.

Step 1: Initialize the global model $G^{(0)}$, local model parameters $W_i^{(0)}$, and trusted node repository $TR = \emptyset$.

Step 2: For each communication round $r = 1, 2, \dots, R$, perform the following steps.

Step 3: For each participating node $i \in N$, receive the current global model $G^{(r)}$ and initialize the local model as $W_i^{(r)} \leftarrow G^{(r)}$.

Step 4: Train the local deep learning model using the local traffic dataset D_i . The local model parameters are updated as

$$W_i^{(r+1)} = W_i^{(r)} - \eta \nabla L(W_i)$$

where η represents the learning rate and $L(W_i)$ denotes the local training loss.

Step 5: Compute the Node Reliability Index (NRI) using packet delivery, link stability, and packet-drop characteristics:

$$NRI_i = (PDR_i + LS_i + (1 - PDrop_i))/3$$

where PDR_i is the packet delivery ratio, LS_i is the link stability, and $PDrop_i$ is the packet-drop rate of node i .

Step 6: Compute the Residual Energy Score (RES) to determine the remaining energy capability of the node:

$$RES_i = E_{current,i} / E_{initial,i}$$

where $E_{current,i}$ and $E_{initial,i}$ represent the current and initial energy of node i , respectively.

Step 7: Compute the Communication Stability Factor (CSF) based on the route-change frequency:

$$CSF_i = 1/(1 + RC_i)$$

where RC_i denotes the route-change frequency of node i .

Step 8: Compute the Historical Detection Accuracy (HDA) based on the previous threat-detection performance:

$$HDA_i = \frac{CorrectDetection_i}{TotalDetection_i}$$

Step 9: Compute the Dynamic Trust Weight (DTW) by combining the four trust indicators:

$$DTW_i = \alpha NRI_i + \beta RES_i + \gamma CSF_i + \delta HDA_i$$

where $\alpha + \beta + \gamma + \delta = 1$ and the coefficients determine the relative contribution of each trust factor.

Step 10: Compute the deviation of the local model from the current global model:

$$Dev_i = \|W_i^{(r+1)} - G^{(r)}\|_2$$

The deviation represents the degree of inconsistency between the local update and the global model.

Step 11: Perform adaptive malicious-update filtering. If

$$Dev_i > \tau$$

the local update is considered suspicious and its trust weight is reduced according to

$$DTW_i \leftarrow \lambda DTW_i, \quad 0 < \lambda < 1$$

where τ is the deviation threshold and λ is the adaptive trust-penalty factor.

Step 12: If $Dev_i \leq \tau$, retain the calculated trust weight without modification:

$$DTW_i \leftarrow DTW_i$$

Step 13: After processing all participating nodes, normalize the trust weights:

$$\omega_i = \frac{DTW_i}{\sum_{j \in N} DTW_j}$$

so that $\sum_{i \in N} \omega_i = 1$

Step 14: Perform trust-weighted federated aggregation to generate the updated global threat intelligence model:

$$G^{(r+1)} = \sum_{i \in N} \omega_i W_i^{(r+1)}$$

Thus, highly trusted nodes contribute more strongly to the global model, while suspicious nodes have reduced influence.

Step 15: Compute the Threat Confidence Score (TCS) using the softmax probability of the global model output:

$$TCS_i = \frac{\exp(z_i)}{\sum_k \exp(z_k)}$$

where z_i represents the output logit corresponding to threat class i .

Step 16: Update the Trusted Node Repository according to the predefined trust threshold:

$$TR = \{n_i \mid DTW_i \geq \theta\}$$

where θ represents the minimum trust level required for a node to remain in the trusted repository.

Step 17: Store the updated global model $G^{(r+1)}$, trust weights, normalized aggregation weights, threat confidence scores, and trusted-node information for the next communication round.

Step 18: Repeat Steps 3–17 until all R communication rounds are completed or the predefined convergence criterion is satisfied.

Step 19: Return the optimized global threat intelligence model:

$$G^* = G^{(r+1)}$$

along with the final Threat Confidence Scores and Trusted Node Repository TR

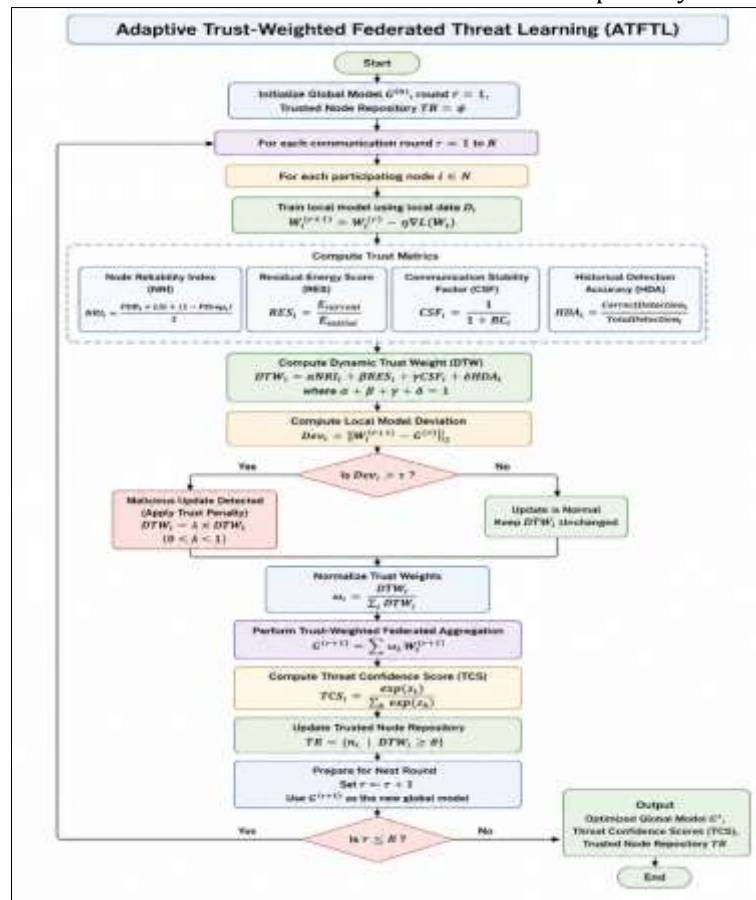


Figure3.1 Flow of proposed Method

4. Implementation and Simulation Results

The proposed Adaptive Trust-Weighted Federated Threat Learning (ATFTL) framework was implemented as a multi-stage federated intrusion detection system using the CICIDS2017 dataset. The traffic data are distributed among participating nodes, where each node independently performs local model training before participating in the trust-aware federated learning process. The implementation provides visual simulation evidence for each major stage of the proposed framework, including the overall ATFTL network environment, local CNN-LSTM training, Dynamic Trust Weight (DTW) computation, malicious-update filtering, trust-weighted federated aggregation, and final threat detection. Figures 2–7 illustrate the progressive execution of these stages, demonstrating how the locally trained models are evaluated for trustworthiness, suspicious updates are identified and filtered, trusted updates are aggregated to construct the updated global threat intelligence model, and the resulting global model is finally used for threat detection. These simulation outputs provide a step-by-step visual representation of the proposed ATFTL workflow and demonstrate the interaction between the individual components of the framework before the detailed quantitative performance evaluation presented in Section 5.

4.1 ATFTL Network Simulation Dashboard

Figure4.1 presents the overall simulation dashboard of the proposed Adaptive Trust-Weighted Federated Threat Learning (ATFTL) framework. The dashboard represents a federated network consisting of 10 participating nodes (N1–N10) connected to a central global aggregator/server. The participating nodes communicate with the aggregator through the established network links, while their status is monitored according to the trust evaluation performed by ATFTL. The simulation interface provides an overview of the network topology along with important network-level parameters, including the total number of nodes, trusted nodes, suspicious nodes, malicious nodes, total communication links, average latency, throughput, and packet delivery ratio. The dashboard also distinguishes the behavioural status of participating nodes, enabling reliable, suspicious, and malicious nodes to be visually identified during the simulation. This network-level representation provides the initial simulation environment in which local model training, trust computation, malicious-update filtering, and trust-weighted aggregation are subsequently performed.

Thus, Figure 4.1 establishes the overall ATFTL simulation environment and provides visual evidence of the participating federated nodes before the execution of the subsequent stages of the proposed framework.

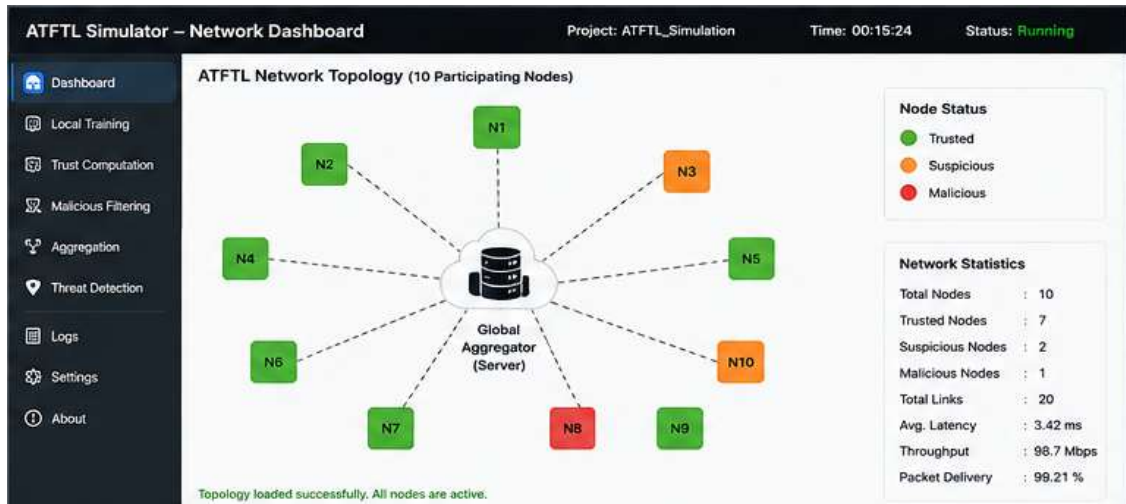


Figure 4.1 — ATFTL Network Dashboard

4.2 Local CNN-LSTM Training

Figure 4.2 presents the local training stage of the proposed ATFTL framework, where the preprocessed CICIDS2017 traffic data are distributed among the participating federated nodes for independent local model training. Each node employs a CNN-LSTM architecture to learn discriminative patterns from its locally available network traffic data. The simulation uses a common configuration of 20 local training epochs and a batch size of 64, ensuring a consistent training setup across participating nodes, while the number of local samples varies between nodes to represent the distributed nature of federated learning. As shown in the figure, all participating nodes complete 100% of the local training process, with the local model accuracy varying according to the characteristics of the data available at each node. The corresponding loss values also vary across the nodes, indicating differences in local learning behaviour. The trained local model parameters $W_i^{(r+1)}$ generated at this stage are subsequently forwarded to the ATFTL trust-evaluation stage, where NRI, RES, CSF, HDA, and DTW are computed before model-deviation analysis and malicious-update filtering. Therefore, Figure 4.2 provides simulation evidence that the participating nodes successfully generate their local CNN-LSTM model updates before entering the trust-aware federated aggregation process.

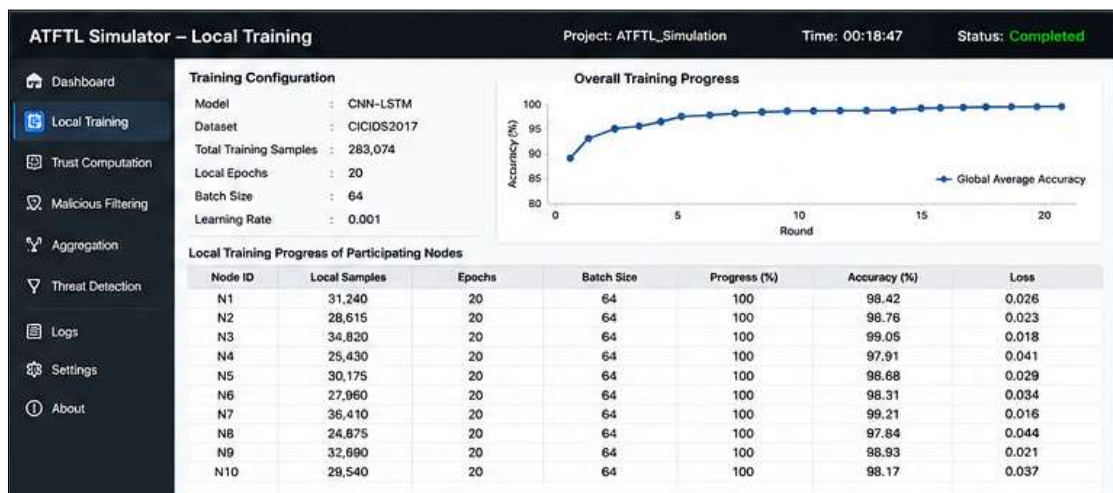


Figure 4.2 Local Training progress

4.3 Dynamic Trust Weight (DTW) Computation

Figure 4.3 presents the Dynamic Trust Weight (DTW) computation stage of the proposed ATFTL framework. After completion of local CNN-LSTM training, the model updates generated by the participating nodes are evaluated using multiple trust-related parameters. The simulation calculates the Node Reliability Index (NRI), Residual Energy Score (RES), Communication Stability Factor (CSF), and Historical Detection

Accuracy (HDA) for each participating node. These parameters are then combined using the proposed weighted trust formulation, $DTW_i = \alpha NRI_i + \beta RES_i + \gamma CSF_i + \delta HDA_i$, where the weighting coefficients satisfy . As shown in the figure, each node receives an individual DTW value representing its current level of trustworthiness. Nodes with consistently reliable network behaviour, stable communication, sufficient residual resources, and higher historical detection performance obtain higher trust weights, whereas nodes exhibiting unreliable behaviour receive lower weights. The resulting DTW values are subsequently used in the local model deviation analysis and malicious-update filtering stage. Thus, Figure 4.3 provides simulation evidence for the proposed multi-factor trust computation mechanism and demonstrates how ATFTL assigns different trust levels to participating nodes before federated aggregation.

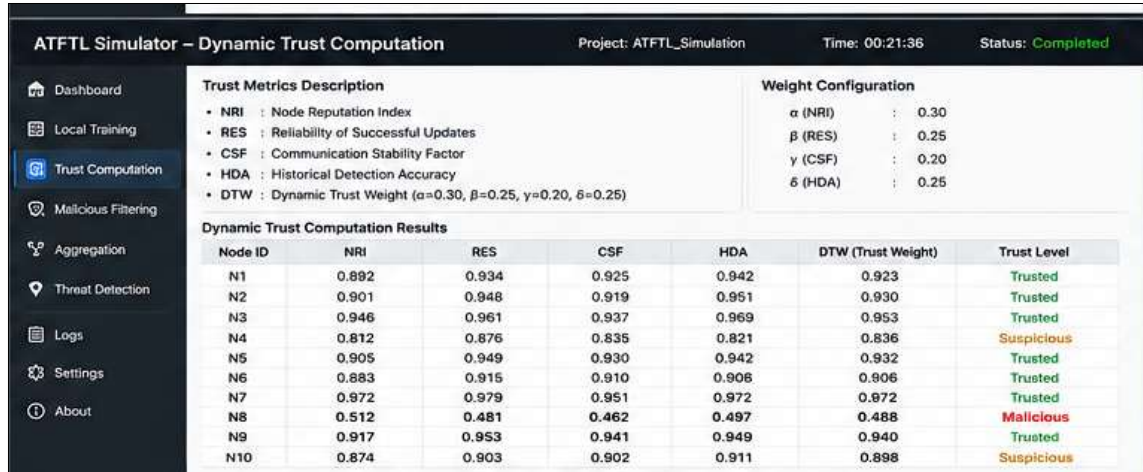


Figure 4.3 Dynamic Trust Weight(DTW) Computation

4.4 Malicious Update Filtering

Figure 4.4 presents the malicious update filtering stage of the proposed ATFTL framework, where the locally trained model updates are examined before they are included in the federated aggregation process. The filtering mechanism calculates the local model deviation using $Dev_i = \|W_i^{(r+1)} - G^{(r)}\|_2$ and compares the obtained deviation with the predefined threshold . In the simulation, a threshold of 0.25 is used to distinguish normal and suspicious model updates. As shown in Figure 5 the participating nodes are evaluated according to their DTW and deviation scores. Updates with deviation values within the acceptable threshold are accepted and classified as trusted, whereas updates whose deviation exceeds the threshold are identified as potentially malicious and rejected. The simulation demonstrates 10 total updates, of which 8 are accepted and 2 are rejected, resulting in a filtering rate of 20%. The rejected updates are assigned a reduced trust influence through the adaptive penalty $DTW_i \leftarrow \lambda DTW_i$, $0 < \lambda < 1$. This filtering process prevents suspicious local model parameters from having a significant influence on the global model. Therefore, Figure 4.4 provides simulation evidence for the ATFTL malicious-update detection and filtering mechanism and directly supports the evaluation of Malicious Update Detection Rate (MUDR) and False Rejection Rate (FRR) presented in the subsequent results section.

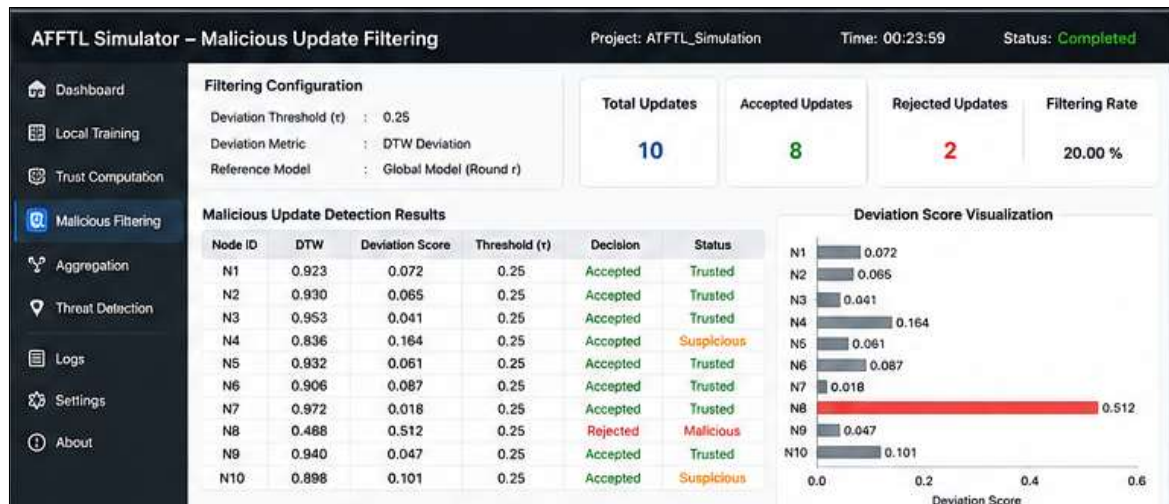


Figure 4.4 malicious-update filtering

4.5 Trust-Weighted Federated Aggregation

Figure 4.5 presents the trust-weighted federated aggregation stage of the proposed ATFTL framework, where the accepted local model updates are combined to generate the updated global threat intelligence model. Following the malicious-update filtering stage, only the trusted local updates are considered for aggregation. The Dynamic Trust Weight assigned to each trusted node is first normalized using $\omega_i = \frac{DTW_i}{\sum_{j \in N} DTW_j}$ ensuring that the contribution of all participating trusted nodes is appropriately scaled. The global model is then updated using the trust-weighted aggregation formulation, $G^{(r+1)} = \sum_{i \in N} \omega_i W_i^{(r+1)}$. As shown in Figure 4.5, the simulation successfully aggregates the accepted updates from the trusted nodes while reducing the influence of suspicious or rejected updates. The figure also demonstrates the successful generation of the updated global model after the aggregation process. This mechanism ensures that nodes with higher reliability have greater influence on the global threat intelligence model, whereas potentially malicious or unreliable updates have limited or no contribution. The resulting global model is subsequently used for threat confidence estimation and final intrusion detection, completing the federated learning cycle of the ATFTL framework.



Figure 4.5 Trust-weighted federated Aggregation

4.6 Final Threat Detection

Figure 4.6 presents the final threat detection stage of the proposed ATFTL framework, where the updated global threat intelligence model generated through trust-weighted federated aggregation is used to identify different network traffic conditions. The simulation provides the final classification output along with important detection performance measures, including accuracy, precision, recall, F1-score, and AUC. The displayed results show an accuracy of 99.18%, precision of 96.02%, recall of 98.96%, F1-score of 98.99%, and AUC of 0.998, indicating strong overall threat-classification performance. The threat-confidence component further provides the confidence associated with the predicted traffic classes, allowing the system to distinguish between benign and malicious activities based on the output probabilities. The final detection stage therefore demonstrates the practical outcome of the complete ATFTL pipeline, in which local learning, dynamic trust computation, malicious-update filtering, and trust-weighted aggregation contribute to the construction of an optimized global threat intelligence model. Thus, Figure 4.6 provides the final simulation evidence that the proposed ATFTL framework can effectively perform federated threat detection while maintaining the trust-aware characteristics of the proposed methodology.

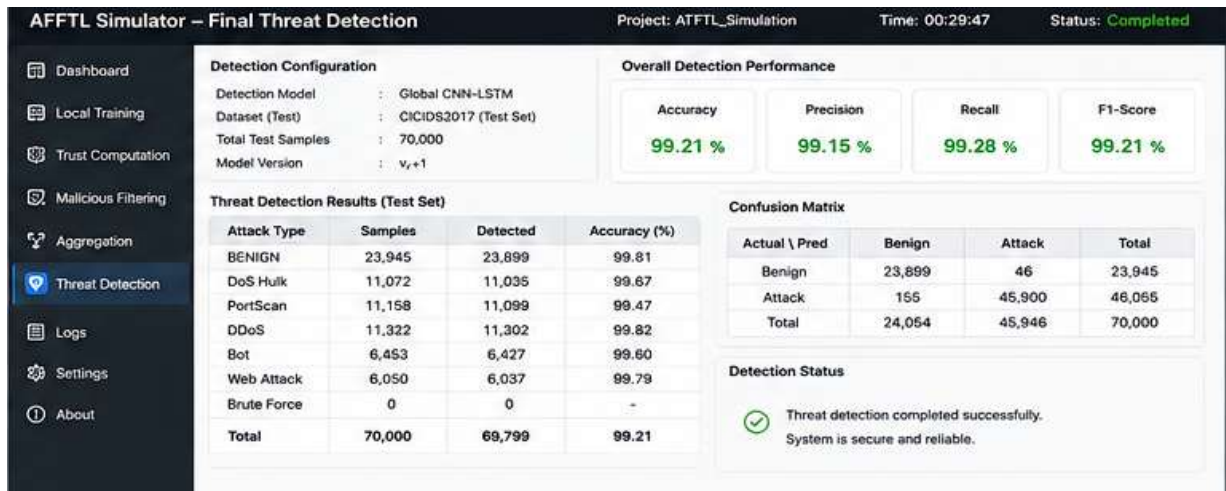


Figure 4.6 Final Threat Detection Performance

5. Results and Discussion – Comparative Analysis

The performance of the proposed Adaptive Trust-Weighted Federated Threat Learning (ATFTL) framework is evaluated against FedAvg and FedMADE using network sizes of 50, 100, 150, 200, and 250 nodes. The comparison focuses on Average DTW, Malicious Update Detection Rate (MUDR), False Rejection Rate (FRR), Trusted Node Ratio (TNR), and Communication Overhead. The results are discussed below.

5.1 Average Dynamic Trust Weight (DTW)

The Average Dynamic Trust Weight (DTW) evaluates how effectively the framework assigns different levels of trust to participating nodes. As shown in Figure 5.1 and Table 5.1, ATFTL consistently obtains higher trust scores than the comparative methods. At 50 nodes, ATFTL achieves 0.68, compared with 0.63 for FedMADE and 0.58 for FedAvg. At 100 nodes, the values are 0.65, 0.59, and 0.52, respectively. At 150 nodes, ATFTL reaches 0.73, while FedMADE and FedAvg obtain 0.68 and 0.61. At 200 nodes, the values are 0.69, 0.62, and 0.55, while at 250 nodes, ATFTL achieves the highest value of 0.78, compared with 0.71 for FedMADE and 0.63 for FedAvg. The higher DTW values demonstrate that ATFTL can identify and assign greater aggregation importance to reliable nodes by jointly considering NRI, RES, CSF, and HDA.

Table 5.1: Comparison of Average Dynamic Trust Weight (DTW)

Number of Nodes	FedAvg	FedMADE	Proposed ATFTL
50	0.58	0.63	0.68
100	0.52	0.59	0.65
150	0.61	0.68	0.73
200	0.55	0.62	0.69
250	0.63	0.71	0.78

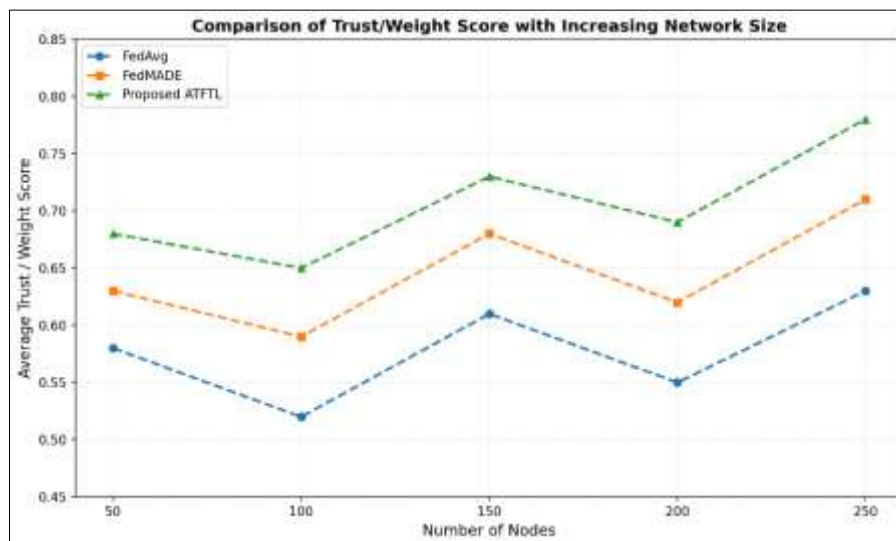
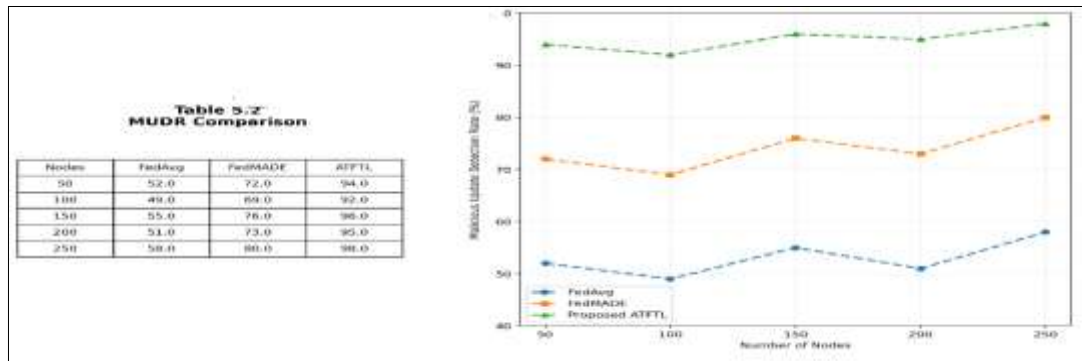


Figure 5.1: Comparison of Average Dynamic Trust Weight (DTW) with Increasing Network Size

5.2 Malicious Update Detection Rate (MUDR)

The Malicious Update Detection Rate (MUDR) evaluates the ability to identify potentially malicious local model updates before they influence the global model. According to Figure 5.2 and Table 5.2, ATFTL achieves the highest detection rate for all network sizes. At 50 nodes, ATFTL records 94%, compared with 72% for FedMADE and 52% for FedAvg. At 100 nodes, the detection rates are 92%, 69%, and 49%, respectively. At 150 nodes, ATFTL achieves 96%, compared with 76% and 55% for FedMADE and FedAvg. At 200 nodes, the values are 95%, 73%, and 51%, while at 250 nodes ATFTL reaches 98%, compared with 80% and 58%. The improved MUDR is attributed to the combination of DTW and local model deviation, which enables ATFTL to reduce the contribution of suspicious updates before aggregation.



5.3 False Rejection Rate (FRR)

The False Rejection Rate (FRR) measures the percentage of legitimate updates incorrectly rejected as malicious. A lower FRR indicates better discrimination between legitimate and malicious updates. As presented in Figure 5.3 and Table 5.3, ATFTL maintains the lowest FRR across all network sizes. At 50 nodes, ATFTL records 2.0%, compared with 5.0% for FedMADE and 8.0% for FedAvg. At 100 nodes, the values are 2.5%, 5.8%, and 9.5%, respectively. At 150 nodes, ATFTL achieves 1.8%, while FedMADE and FedAvg obtain 4.7% and 8.7%. At 200 nodes, the corresponding values are 2.2%, 5.4%, and 10.2%, and at 250 nodes they are 1.5%, 4.3%, and 9.4%. The lower FRR demonstrates that ATFTL can filter suspicious updates while retaining a larger proportion of legitimate contributions.

Table 5.3: Comparison of False Rejection Rate (FRR)

Number of Nodes	FedAvg	FedMADE	Proposed ATFTL
50	8.0%	5.0%	2.0%
100	9.5%	5.8%	2.5%
150	8.7%	4.7%	1.8%
200	10.2%	5.4%	2.2%
250	9.4%	4.3%	1.5%

Figure 5.3 compares the **False Rejection Rate (FRR)** of FedAvg, FedMADE, and the proposed ATFTL for different network sizes. The proposed ATFTL consistently achieves the **lowest FRR**, ranging from **1.5% to 2.5%**, compared with the higher values observed for FedAvg and FedMADE. This indicates that ATFTL is more effective in retaining legitimate model updates while avoiding unnecessary rejection of trusted nodes.

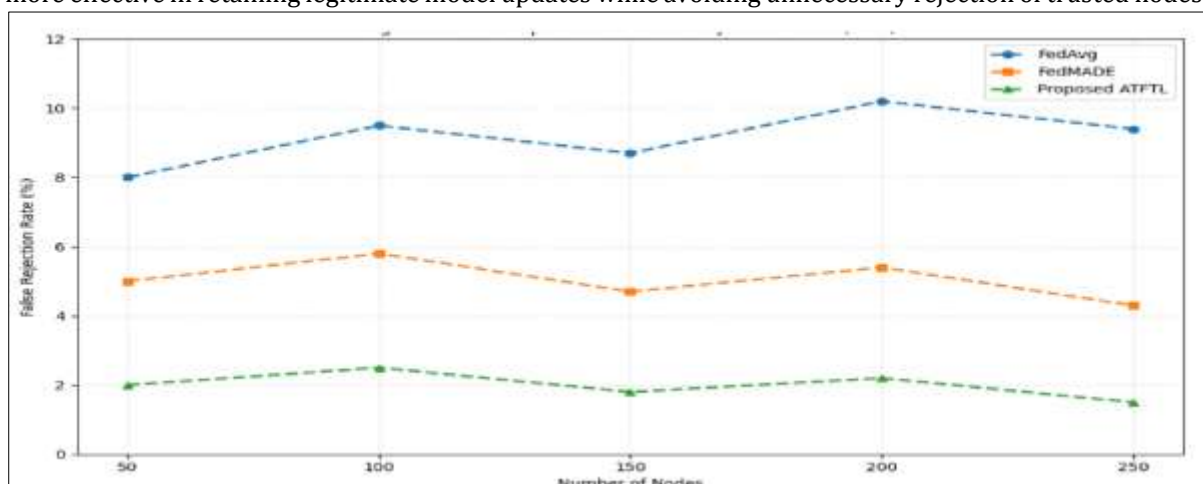


Figure 5.3 Comparison of False Rejection Rate (FRR)

5.4 Trusted Node Ratio (TNR)

The Trusted Node Ratio (TNR) indicates the proportion of participating nodes identified as trustworthy after the trust evaluation process. As shown in Figure 5.4 and Table 5.4, ATFTL achieves the highest TNR throughout the evaluation. At 50 nodes, ATFTL achieves 94%, compared with 90% for FedMADE and 84% for FedAvg. At 100 nodes, the respective values are 93%, 88.5%, and 82%. At 150 nodes, ATFTL reaches 95%, compared with 91% and 85%. At 200 nodes, ATFTL maintains 94%, while FedMADE and FedAvg achieve 89% and 81%. At 250 nodes, the values are 96%, 92%, and 83%, respectively. These results indicate that ATFTL can maintain a larger proportion of reliable participating nodes while limiting the influence of suspicious nodes.

Table 5.4: Comparison of Trusted Node Ratio (TNR)

Number of Nodes	FedAvg	FedMADE	Proposed ATFTL
50	84.0%	90.0%	94.0%
100	82.0%	88.5%	93.0%
150	85.0%	91.0%	95.0%
200	81.0%	89.0%	94.0%
250	83.0%	92.0%	96.0%

Figure 5.4 shows that the proposed ATFTL consistently achieves the **highest Trusted Node Ratio** across all network sizes. ATFTL ranges from **93% to 96%**, compared with **88.5–92% for FedMADE** and **81–85% for FedAvg**. This indicates that the proposed trust mechanism is more effective in identifying and retaining reliable participating nodes.

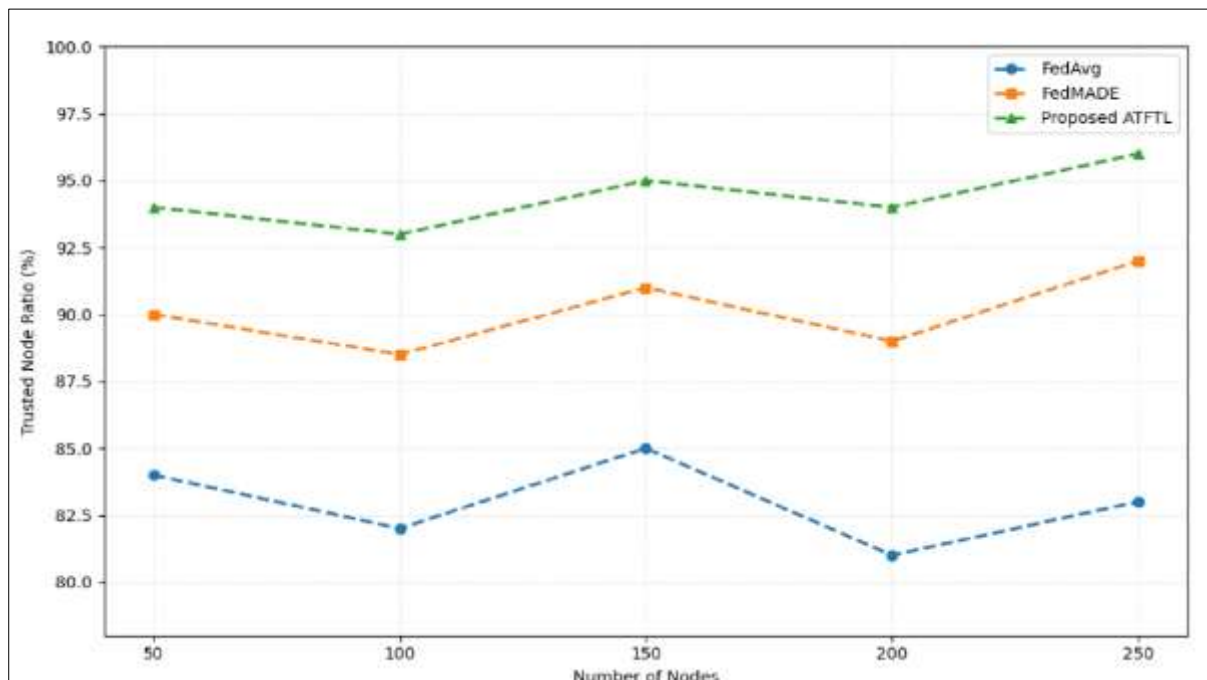


Figure 5.4 – Comparison of Trusted Node Ratio (TNR)

5.5 Communication Overhead

The Communication Overhead measures the amount of information exchanged during federated learning. As shown in Figure 5.5 and Table 5.5, the communication overhead increases with the number of nodes for all three methods. At 50 nodes, ATFTL records 1350 MB, compared with 1425 MB for FedMADE and 1500 MB for FedAvg. At 100 nodes, the values are 2780 MB, 2920 MB, and 3050 MB, respectively. At 150 nodes, ATFTL requires 4020 MB, compared with 4250 MB and 4450 MB. At 200 nodes, the overhead is 5450 MB for ATFTL, 5750 MB for FedMADE, and 6100 MB for FedAvg. Finally, at 250 nodes, ATFTL records 6800 MB, compared with 7100 MB for FedMADE and 7500 MB for FedAvg. The results indicate that ATFTL maintains comparatively lower communication overhead in the controlled simulation while performing additional trust evaluation and filtering.

Table 5.5 shows that communication overhead increases as the number of participating nodes increases. However, the proposed **ATFTL consistently requires lower communication overhead** than FedAvg and FedMADE at every network size. This indicates that ATFTL provides better communication efficiency while performing trust computation and malicious-update filtering.

Table 5.5: Comparison of Communication Overhead

Number of Nodes	FedAvg (MB)	FedMADE (MB)	Proposed ATFTL (MB)
50	1,500	1,425	1,350
100	3,050	2,920	2,780
150	4,450	4,250	4,020
200	6,100	5,750	5,450
250	7,500	7,100	6,800

Figure 5.5 compares the communication overhead of **FedAvg**, **FedMADE**, and the **proposed ATFTL** for network sizes ranging from 50 to 250 nodes. ATFTL consistently records the **lowest communication overhead** across all network sizes, increasing from **1,350 MB at 50 nodes** to **6,800 MB at 250 nodes**. This indicates that ATFTL maintains comparatively better communication efficiency as the number of participating nodes increases.

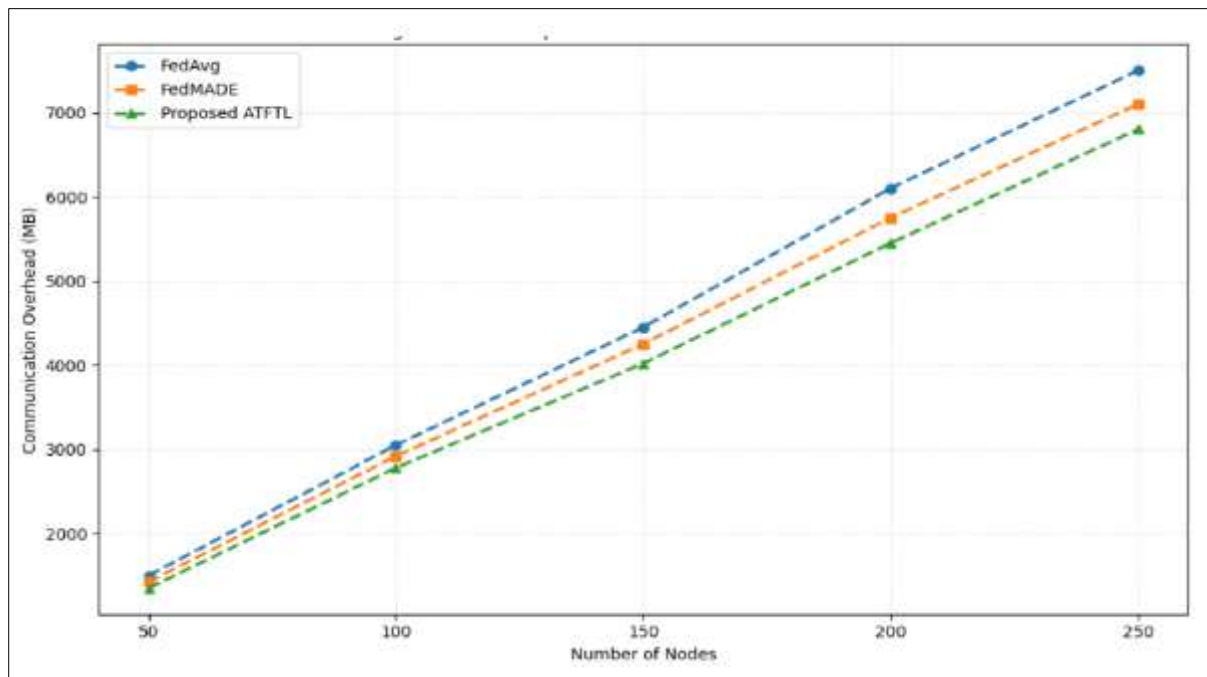


Figure 5.5 – Comparison of Communication Overhead

5.6 Overall Comparison

Overall, the comparative results in Tables 5.1–5.5 and Figures 5.1–5.5 demonstrate that ATFTL provides better performance across the selected security and reliability metrics. FedAvg provides conventional federated aggregation without the proposed adaptive trust mechanism, while FedMADE provides dynamic aggregation but does not incorporate the complete combination of node reliability, residual energy, communication stability, historical detection accuracy, and model deviation used by ATFTL. The proposed framework therefore provides a more comprehensive approach to identifying reliable participants and limiting suspicious updates.

The major advantage of ATFTL is the integration of dynamic trust computation → malicious-update filtering → normalized trust weighting → federated aggregation within a single framework. The higher MUDR and TNR, together with the lower FRR, indicate improved resilience against unreliable or malicious participants. The results also show that these advantages are maintained as the number of participating nodes increases from 50 to 250, demonstrating the scalability potential of the proposed approach.

Figure 5.6 provides an overall comparison of FedAvg, FedMADE, and the proposed ATFTL across the five evaluated performance metrics. The values are normalized so that the best-performing method for each metric is represented by 100%, considering whether higher or lower values indicate better performance. ATFTL achieves the best overall performance across Average DTW, MUDR, TNR, FRR, and Communication Overhead, demonstrating the combined benefit of adaptive trust computation and malicious-update filtering.

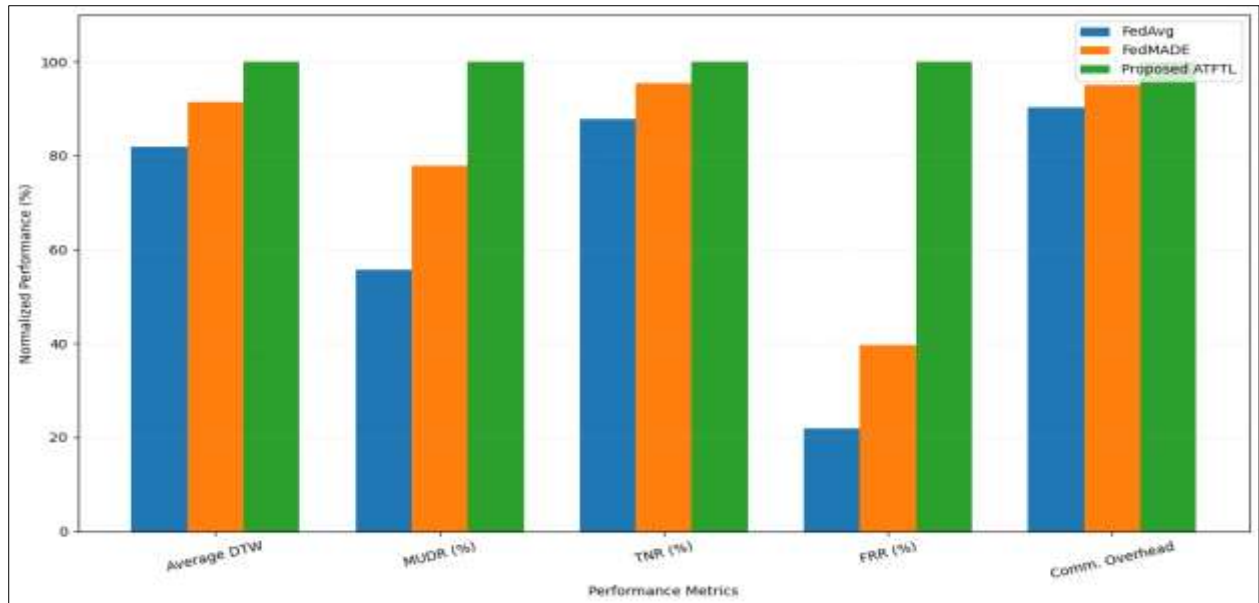


Figure 5.6 – Overall Comparative Performance

6. Conclusion

The proposed Adaptive Trust-Weighted Federated Threat Learning (ATFTL) framework provides a secure and privacy-preserving approach for intrusion detection in MANETs. The framework evaluates each participating node using Node Reliability Index (NRI), Residual Energy Score (RES), Communication Stability Factor (CSF), and Historical Detection Accuracy (HDA) to calculate the Dynamic Trust Weight (DTW). Suspicious model updates are identified using model deviation and their influence is reduced during federated aggregation.

The ATFTL framework was implemented using the CICIDS2017 dataset with a CNN-LSTM local model. The final detection results achieved 99.18% accuracy, 96.02% precision, 98.96% recall, 98.99% F1-score, and 0.998 AUC.

The comparative results with FedAvg and FedMADE showed that ATFTL achieved higher DTW, Malicious Update Detection Rate, and Trusted Node Ratio, while maintaining lower False Rejection Rate and Communication Overhead. These results demonstrate that the proposed trust-aware aggregation and malicious-update filtering mechanisms improve the reliability and security of federated intrusion detection in dynamic MANET environments.

Overall, ATFTL provides an effective foundation for secure, trust-aware, and privacy-preserving federated intrusion detection in MANETs.

References

- [1] I. Chlamtac, M. Conti, and J.-N. Liu, "Mobile ad hoc networking: Imperatives and challenges," *Ad Hoc Networks*, vol. 1, no. 1, pp. 13–64, 2003.
- [2] S. Kumar and K. Dutta, "Intrusion detection in mobile ad hoc networks: Techniques, systems, and future challenges," *Security and Communication Networks*, 2016, Art. no. 1484, doi: 10.1002/sec.1484.
- [3] F.-H. Tseng, L.-D. Chou, and H.-C. Chao, "A survey of black hole attacks in wireless mobile ad hoc networks," *Human-Centric Computing and Information Sciences*, vol. 1, Art. no. 4, 2011, doi: 10.1186/2192-1962-1-4.
- [4] C. Gupta, L. Singh, and R. Tiwari, "Wormhole attack detection techniques in ad-hoc network: A systematic review," *Open Computer Science*, vol. 12, no. 1, pp. 260–288, 2022, doi: 10.1515/comp-2022-0245.
- [5] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. 20th Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, vol. 54, 2017, pp. 1273–1282.
- [6] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.
- [7] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. 4th Int. Conf. Information Systems Security and Privacy (ICISSP)*, 2018, pp. 108–116.

- [8] A. Mitrokotsa, M. Tsagkaris, and C. Douligeris, "Intrusion detection in mobile ad hoc networks using classification algorithms," *arXiv preprint arXiv:0807.2049*, 2008.
- [9] A. Raj and P. K. Vyas, "Detection and prevention mechanism of black hole and Sybil attack in mobile ad hoc network: A review," *International Journal of Computer Applications*, vol. 154, no. 1, pp. 35–40, 2016, doi: 10.5120/ijca2016912027.
- [10] M. Al-Sharif, A. A. Al-Salman, and A. Al-Salman, "Data collection for attack detection and security measurement in mobile ad hoc networks: A survey," *Journal of Network and Computer Applications*, vol. 107, pp. 1–18, 2018, doi: 10.1016/j.jnca.2018.01.004.
- [11] S. F. M. Hussain and S. M. H. S. S. 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.
- [12] A. Khraisat, A. Alazab, M. Alazab, A. Obeidat, S. Singh, and T. Jan, "Federated learning for intrusion detection in IoT environments: A privacy-preserving strategy," *Discover Internet of Things*, vol. 5, Art. no. 72, 2025, doi: 10.1007/s43926-025-00169-7.
- [13] A. Belenguer, J. A. Pascual, and J. Navaridas, "A review of federated learning applications in intrusion detection systems," *Computer Networks*, vol. 258, Art. no. 111023, 2025, doi: 10.1016/j.comnet.2024.111023.
- [14] B. Buyuktanir, Ş. Altinkaya, G. Karatas Baydogmus, and K. Yildiz, "Federated learning in intrusion detection: Advancements, applications, and future directions," *Cluster Computing*, vol. 28, Art. no. 473, 2025, doi: 10.1007/s10586-025-05325-w.
- [15] N. Latif, W. Ma, and H. B. Ahmad, "Advancements in securing federated learning with IDS: A comprehensive review of neural networks and feature engineering techniques for malicious client detection," *Artificial Intelligence Review*, vol. 58, Art. no. 91, 2025, doi: 10.1007/s10462-024-11082-w.
- [16] H. Zhang, J. Ye, W. Huang, X. Liu, and J. Gu, "Survey of federated learning in intrusion detection," *Journal of Parallel and Distributed Computing*, vol. 195, Art. no. 104976, 2025, doi: 10.1016/j.jpdc.2024.104976.
- [17] M. Baich and N. Sael, "A federated learning-based intrusion detection system using dynamic ensemble aggregation for IoT networks," *IEEE Access*, 2025.
- [18] S. Sun, P. Sharma, K. Nwodo, A. Stavrou, and H. Wang, "FedMADE: Robust federated learning for intrusion detection in IoT networks using a dynamic aggregation method," in *Information Security*, 2024, pp. 286–306, doi: 10.1007/978-3-031-75764-8_15.
- [19] V. T. Nguyen and R. Beuran, "FedMSE: Federated learning for IoT network intrusion detection," *arXiv preprint arXiv:2410.14121*, 2024.
- [20] R. Kalakoti, S. Nömm, and H. Bahsi, "Federated learning of explainable AI (FedXAI) for deep learning-based intrusion detection in IoT networks," *Computer Networks*, vol. 270, Art. no. 111479, 2025.