University of Sindh Journal of Information and Communication Technology (USJICT)
Volume 9, Issue 2, 2025
ISSN-E: 2523-1235, ISSN-P: 2521-5582
Website: http://sujo.usindh.edu.pk/index.php/USJICT/

An Enhanced Lightweight Trust Framework for Malicious Node Detection in Edge-Enabled IIoT Systems Networks

Atif hayat[1], Dr.Asim Zeb[1],Dr. Muhammad Shehzad Khan[1]*, Dr.Muhammad Naeem[1]
Department of Computer Science, Abbottabad University of Science & Technology, 22010 Havelian, Pakistan

atifhayatq6@gmail.com, asim.zeb@aust.edu.pk, me_shehzadkhan@yahoo.com, naeem@aust.edu.pk

Abstract: The Industrial Internet of Things (IIoT) is a technology that allows efficient communication between interconnected devices, sensors, and edge nodes, making advanced industrial automation possible. But edge devices with limited resources are prone to security issues like malicious node behavior, fake recommendations, and attacks on the trust mechanisms. The existing lightweight trust management schemes are mainly based on direct interaction and static trust assessment, which is not suitable for dynamic environments. In this paper, a new lightweight trust management system for enhancing the trustworthiness of IIoT networks is proposed. The proposed approach is a hybrid trust evaluation approach combining direct trust evaluation based on observation and indirect trust evaluation based on recommendation feedback. In addition, for achieving continuous adaptation of trust values based on nodes' behaviors, a dynamic trust adaptation strategy is introduced. The results of the simulations under various network conditions and malicious node densities demonstrate that the proposed framework offers an effective improvement of the accuracy of the Trust Assessment, malicious node detection and defense against dishonest recommendations. The model is stable with as many as 50% malicious nodes in the network and has low computational and communication overhead. The framework offers a trustworthy edge computing trust management solution that is secure, scalable and efficient for IIoT environments.

Keywords: Direct trust, Indirect trust, Dynamic trust, Industrial Internet of Things, Edge devices;

I. INTRODUCTION

Since the Internet is a rapidly evolving technology in the modern era, it has enabled the widespread development of the Internet of Things (IoT) paradigm. IoT refers to a network of interconnected physical devices that communicate and exchange data over the Internet [1]. These devices are capable of sensing their environment, collecting real-time information, and sharing it with other systems to enable intelligent decision-making [2]. IoT supports seamless communication between heterogeneous devices anytime and anywhere, improving connectivity and service efficiency [3], [4].

Industrial Internet of Things (IIoT) brings IoT to industry, with the aim of smart manufacturing, predictive maintenance and industrial automation. The concept of IIoT combines the concepts of edge computing and distributed processing architectures, resulting in better operational efficiency [5]. Edge nodes are essential for industrial applications due to their ability to process data in real time, near the data source, thus minimizing latency [6].

But this is an extension of devices creates important security and trust issues. In the context of IIoT, reliable and secure communication and data processing between edge nodes is crucial. Trust is a critical aspect that has an important role in making interactions safe in a system by checking the reliability and actions of the network objects [7]. It provides a way to deliver safe and efficient services in distributed systems [8]–[10].

There are two types of traditional trust management: centralized and decentralized. There are two types of systems: centralized systems, where trust evaluation is done by a single entity, and decentralized systems, where trust evaluation is performed among several nodes [11, 12]. These methods are used to identify the nodes that might be malicious or unreliable in an IoT network.

To boost efficiency and security in the IIoT environments, LightTrust is a lightweight trust management system that has been designed. It uses a time and event-driven model for trust computation to reduce the computational cost. The framework contains information collection, trust computation, dissemination and trust retention processes [13]

The proposed framework is different from the current trust models in the following ways:

- A hybrid trust aggregation mechanism for integration of direct trust and indirect recommendation trust. Implementation of dynamic trust adaptation to continuously update trust value using node behavior.

- Improved Dishonest Recommendation Attack and Trust Manipulation Protection. Lightweight implementation which can be adapted to resource constrained edge devices.
- Node secure authentication mechanism using trust certificate.
- Integration of direct trust and indirect recommendation trust through a hybrid trust aggregation mechanism.
- Introduction of dynamic trust adaptation to continuously update trust values according to node behavior.
- Enhanced resistance against dishonest recommendation attacks and trust manipulation.
- Lightweight implementation suitable for resource-constrained edge devices.
- Trust certificate generation mechanism for secure node authentication.is required.

The major findings of this study are summarized below:

- Designing a lightweight and scalable trust management system for edge environments in IIoT.Heterogeneous trust assessment based on both direct and indirect knowledge of trust.
- Adaptation mechanism for dynamic trust: detect behavioural changes and counteract trust manipulation attacks.
- Better malicious node detection and energy and computation efficiency.
- Extensive performance analysis with different network conditions and node densities of malicious nodes.

A. Research Gap

Although the trust management works for the IIoT systems are improved, there were some drawbacks:

- The majority of existing lightweight trust models, like LightTrust, are based primarily on direct interaction and static trust computation.
- Most of the trust frameworks are ineffective in preventing false recommendation attacks and trust manipulation attacks.
- There are several strategies that introduce too much communication and computation overhead, thus not suitable for the resource-constrained edge devices.
- Current schemes show poor trust accuracy in highly dynamic environments in industry.
- Existing trust models are not good enough at handling the changing behaviors of nodes over time.

Hence, a lightweight, adaptive and hybrid trust management mechanism that can accurately identify malicious nodes in dynamic IIoT setting with low overhead Novelty of Proposed Work

II. RELATED WORK

Trust is a fundamental requirement for ensuring security in Industrial Internet of Things (IIoT) environments. It is the degree of uncertainty and trust of a system or node, in interaction [14]. In fact, there are interconnected industrial systems in use and therefore, trust is crucial for secure communications and system stability [15].

Defining and measuring trust is challenging since it is comprised of a variety of factors including security, privacy, reliability, availability and integrity. It also is dynamic and context-dependent, which makes it hard to model generally.

Many security methods, such as cryptographic and authentication methods are commonly used to secure IoT communications to guarantee confidentiality and data integrity [16]. These techniques however are not enough to identify behavioral anomalies and insider attacks for dynamic environments.

Several trust evaluation models have been proposed to tackle these limitations. The Efficient Trust Evaluation Strategy (ETES) is based on metrics like packet forwarding ratio and reliability to evaluate node behavior. It is used to improve the computation of trust based on Dempster–Shafer (D–S) theory of combining direct and indirect evidence [17].

Past interactions and neighbor feedback are used to calculate trust using the Subjective Model of Trustworthiness Evaluation (SMTE). However it has storage and computation overheads, yet it is capable of detecting malicious nodes [18].

Group-based Trust Management Strategy (GTMS) is one of the methods used to reduce the energy consumption in WSN by dividing the nodes into two classes, namely trustworthy and malicious nodes. However, this could cause a delay in communication in large scale networks [19].

TM-DTN enhances latency, accuracy and QoS to increase trust evaluation in delay-tolerant networks (DTN). However, its resistance to the advanced attacks still needs to be validated [20].

The QoS-Social Trust models combine social relationships and QoS metrics to determine the trust of IoT networks. These models have enhanced adaptability but the models still need to be further tested in adversarial conditions [21].

ScaleTM-IoT offers a scalable trust model with social interaction between nodes, which guarantees accurate trust

convergence. But in resource limited environments, privacy issues emerge [22].

To enhance reliability in IoT systems, Sub Model-IoT develops subjective trust estimation based on past behavior and recommendations of trusted nodes.

DTF are useful for Communities of Interest (CoI) in a scalable, flexible, and resilient way within a heterogeneous IoT environment [24].

LightTrust is a lightweight trust system that leverages centralized trust agents and edge nodes to efficiently compute trust with low energy consumption [25].

Recent studies have demonstrated that machine learning and deep learning-based AI trust models have been shown to greatly enhance anomaly detection, trust prediction, and adaptive security in IIoT systems [26].Overall, literature indicates that hybrid, dynamic, and lightweight trust management approaches are essential for secure and efficient IIoT systems.

III. METHODOLOGY

A. Proposed Mechanism Working

Trust Development (Trust Developer) (Trust Agent) and Trust Management (Trust Manager) comprises on the recommended approach. The trust agent can help build a reliable conduit for information transfer between several entities.

In addition, it records the total amount, appends it to the database, adds it to the trust developer's trust values, and gives it a unique ID. Six criteria are used by Trust developers who build trust between heterogeneous nodes: compatibility, cooperation, delivery ratio, correctness, reliability, and suggestions.

It is possible to establish trust through both direct and indirect observations. A node can interact directly with other nodes in the network and exchange messages after instantaneous observation. In contrast, indirect observations allow information to be sent between two nodes through a connection based on suggestions made by a third node based on the node's standing. A time-based certificate, which might have a long or short duration, is granted to the evaluated node based on the predetermined values of trust. The computed value supplied by the trustee node and the trust developer, which was submitted by the trustor, is the only foundation upon which certifications may be granted. Every node's trust value is kept in the database of the trust agent.

B. Establishing Trust Parameters

In order for increasing the SLA functioning and establish a trustworthy channel and environment for IoT devices, two more factors—accuracy and reliability—are chosen in addition to the four pre-existing trust factors—recommendations, compatibility, delivery ratio, and cooperation.

The properties of the matching parameters are represented in a way that a node can set up trust and stay calmly with every other. Cooperation is a potential of a node to engage socially with other nodes in the network. One of the necessary aspects of the supplier relationship is the hub and the capability of social cooperation. The effectivity of the ratio of sent and obtained content material among two interactive devices can be evaluated through the transmission ratio.

Compatibility trust parameter is based on a node's ability to adapt, i.e., if it can live with other nodes under specific conditions. It indicates whether or not the node's goal of building interpersonal trust has been accomplished. A node's accuracy can be judged more reliable than other nodes' if it consistently gives accurate information. Communication can be made more dependable and trustworthy by include correctness as an element in the trust management system. Reliability of a node is defined as its ability to perform their assigned tasks consistently and without errors.

C. Parameter Computation Process

To decide whether two interacting nodes are trustworthy, the algorithms are taken into account. The underlying values of the trustee are demonstrated to be those of the trustor through these methods. According to their level of reliability, the trustee is rated in this scenario as a node that requests services from the trustor. Additionally, it bases its conclusion on the predetermined algorithms and the defined trust values.

The proposed strategy aims to minimize energy usage by distributing certificates to reliable nodes and holding them for a predefined period of time so that their trust values can be added to later. The method makes use of historical trust levels to improve scalability and reduce the likelihood that on-off attacks will be successful. In the process, energy consumption is maximized and a dependable communication infrastructure is maintained.
In order to expedite additional aggregation and propagation, it verifies dependable nodes and tracks their trust levels for a certain duration. Utilizing the suggested approach also saves energy. By using previous trust levels, scaling is enhanced and on-off attack success rate is reduced.

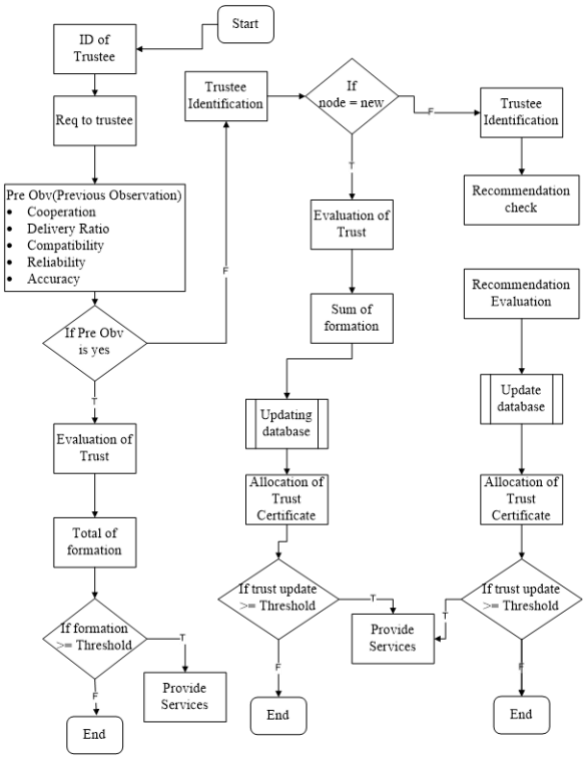

Figure 1. Model's flowchart

Algorithm 1 :Assessing trust through direct observation.

1: procedure TRUST ASSESSMENT (a→b)

2: b_{id}. **The Trustee's identity.**

3: b_{req}→a . **Trustee's Request to Trustor**

4: Pre_{ob}: a→ b [$com_{a\to b}$, $coop_{a\to b}$, $dlr_{a\to b,}$ $acc_{a\to b}$,$rel_{a\to b}$]

5: if (req_{ob}: → b = = true)

6: Move to Step-9

7: else

8: Move to Algorithm 2

9: Eva_{Trust} : a→ b [$com_{a\to b}$, $coop_{a\to b}$, $dlr_{a\to b,}$ $acc_{a\to b}$,$rel_{a\to b}$]

10: $\sum_{0.0}^{1.0} T_{a\to b}^{form}$ = ∑ [$com_{a\to b}$, $coop_{a\to b}$, $dlr_{a\to b,}$ $acc_{a\to b}$,$rel_{a\to b}$]

11: if ($T_{a\to b}^{form}$>= threshold) .**Comparing the Threshold.**

12: Provision of Service;

13: else

14: rejected; **Takingdecisions by using a threshold.**

15: End

D. Trust degree assessment

First step in direct trust assessment is to classify each trustee based on its unique ID. To assist a trustor in determining the trust level, here's an example from Algorithm 1 of how direct confidence observations work: The trustee sets up an account in the network and requests the trustor for help.

Since b denotes the trustee and id denotes its identity, bid in this case shows the trustee's identity when it requests services from the trustor. The method breq→a describes the trustee's request, where req stands for the trustee's request and a for the trustor. When a trustee requests assistance from a trustor, the system gets going. The assessing method starts by initializing the trustee node and its network identity, then uses compatibility, collaboration, delivery ratio, correctness, and reliability as criteria for trust as stated in Equation (1) to verify the level of mistrust in the trustee.

$$Pre_{ob}: a\to b\ [com_{a\to b}, coop_{a\to b}, dlr_{a\to b,} acc_{a\to b}, rel_{a\to b}] \quad (1)$$

Equation (1) denotes the following: *a* is truster and *b* is trustee; *com ,rel, acc, coop* and *dlr,* stand for compatibility, reliability, accuracy , cooperativeness and delivery ratio; and Pre_{ob} stands for previous observation.

$$\text{if (reqob : a} \to \text{b == Yes)} \quad (2)$$

Equation (2) shows the preliminary observations that the trustees *b* have to make for the trustor *a*. If the required observations are all yes, trustee *a* will evaluate trustee *b*; if not, recommendations are used to determine the trust value.

$$Eva_{Trust}: a\to b\ [com_{a\to b}, coop_{a\to b}, dlr_{a\to b,} acc_{a\to b}, rel_{a\to b}] \quad (3)$$

Equation (3) explains the evaluation procedure for trustees *a* through *b*. In this equation, *Eva* represents the trust evaluation.

$$\sum_{0.0}^{1.0} T_{a\to b}^{form} = \sum [com_{a\to b}, coop_{a\to b}, dlr_{a\to b,} acc_{a\to b}, rel_{a\to b}] \quad (4)$$

In equation (4), *T* denotes the trust value; *form* describes the process by which a trust is established; and *a* is trustor and *b* is trustee, respectively.
As indicated by equation (5), trustor*a* offers services to trustee*b*,and if the ultimate value of the trust becomes equal to or exceeds the predetermined threshold value, the trustor begins communication.

$$If(T_{a\to b}^{form} >= threshold) \quad (5)$$

E. Process of developing trust:

Using the trustor's unique ID, algorithm (2) is initialized with the trustee's ID when the trustee seeks services from the trustor. In this instance, b_{id} defines the trustee's identity. The algorithm is initiated and ID provides the trustee with a distinct identity that the trustor can identify if they want to receive services.

Algorithm (2) provides a sequential process for figuring out a person's level of trust. Algorithm (2)'s third step uses equation *if(b==new)* to determine whether the trustee is a recent network addition. In case the trustee is not well-known, trusts are evaluated by means of an algorithm (3). If not, trusts are tested against the trustee.

Algorithm 2: Parameter Formulation with Absolute Trust

1: procedure TRUST ASSESSMENT (a→b)

2: b_{id} . **The Trustee's identity.**

3: if(b == new) **Checking for New Node**

4: Move to 7

5: else

6: Move to Algorithm - 3

7: Eva_{Trust} : a → b [$com_{a\to b}$, $coop_{a\to b}$, $dlr_{a\to b,}$ $acc_{a\to b}$,$rel_{a\to b}$]

8: $\sum_{0.0}^{1.0} T_{a\ \to\ b}^{form}$ = ∑ [$com_{a\to b}$, $coop_{a\to b}$, $dlr_{a\to b,}$ $acc_{a\to b}$,$rel_{a\to b}$]

9: D^{update}: $T_{a\ \to\ b}^{form} = T_{a\ \to\ b}^{update}$

10: TB^{cert} : allocate(a→b) . **Certification**

11: if ($T_{a\ \to\ b}^{update}$>= threshold) **Decision Making**

12: Provision of Service;

13: Refusal;

14: End

An example of how trust values are assessed is given by equation (6). The terms evaluation (e), combination (∑), compatibility (com), cooperation (coop), and delivery ratio (dlr) are represented in this equation, respectively. Accuracy (acc) and dependability (reli) are also included. Equation (7), where f represents the trust formulation, illustrates the process of establishing trust by including the previously established trust parameters.

Eva_{Trust} : a → b [$com_{a\to b}$, $coop_{a\to b}$, $dlr_{a\to b,}$ $acc_{a\to b}$,$rel_{a\to b}$] (6)

$Trust_f$: $\sum_{0.0}^{1.0} T_{a\ \to\ b}^{form}$ = ∑ [$com_{a\to b}$, $coop_{a\to b}$, $dlr_{a\to b,}$ $acc_{a\to b}$,$rel_{a\to b}$] (7)

Updating of database is shown in Equation (8). D refers to the database. In $T_{a\ \to\ b}^{form}$, T represents for trust, *form* represents how the trust is formed from trustor a to trustee b, and T^{update} shows on the other hand, denotes the process by which trust wa s modified or altered.

$$D^{update}: T_{a\ \to\ b}^{form} = T_{a\ \to\ b}^{update} \quad (8)$$

After TB^{cert} Equation (10) is used, this assigned a certification to trustee b based on time. In Equation (9), formulating process of the decision is done using the updated threshold value. If the set limit is met, the trustor will begin communicating and provide trustee services..

The node is not permitted to participate in communications when there is less trust. It is also blocked for nine hundred seconds. Additionally, the technique gives the node a default degree so that it can regain greater trust. After being blocked three times, a node is permanently barred from the Internet of Things ecosystem and is unable to re-enter.

$$TB^{cert} : \text{allocate}(a \to b) \quad (9)$$

$$\text{if } (T_{a\ \to\ b}^{update} >= \text{threshold}) \quad (10)$$

F. Assessing indirect trust using recommendations:

The trust evaluation process involves network nodes for exchanging requests and proposals. Algorithm (3) shows the entire flow process.

Algorithm 3 : Indirect Trust Evaluation based onRecommendation

1: procedure TRUST ASSESSMENT (a→b)

2: b_{id} . **The Trustee's identity.**

3: $Recom^{check}$ [a→b]

4: $Recom_{c\ \to\ gth}^{eva}$: [$u_{g1\to c} + u_{g2\to c}$+------+ $u_{gn\to c}$]

5: $Recom_{c\ \to\ gth}^{form}$ = $\sum_{0.0}^{1.0}$($u_{g1\to b} + u_{g2\to b}$ +------+ $u_{gn\to b)}$)

6: D^{update} : $Recom_{e\ \to\ gth}^{form} = T_{a\ \to\ b}^{update}$

7: TB^{cert} : assign(a→b)

8: if ($T_{a\ \to\ b}^{update}$>= threshold)

9: Provision of Service;

10: else

11: Decline;

12: End

Depend on their previous know-how and experience with trustees, neighbouring nodes propose that services should be provided to trustee for. To confirm trustor *a*'s recommendation for trustee *b*, utilized in equation (11a).

Equation (11b) describes the instructions gained from nearby nodes to evaluate the reliability of trustees.
The total number of recommendations from nearby nodes d eter-
mines how much trust they have. These recommendations must be within a specific range of values. *Recom* is recommendations, *eva* represents evaluation, *g* means the neighbouring node, and the sum of neighbours is denoted by *th*. Equation (11c) calculates the trust value for the trustees based on the compiled recommendations. Here, ∑ represents the total of all recommendations received from the neighboring nodes.
The total number of recommendations from nearby nodes decides the level of trust. The recommendations
are values that should be within a certain range to be considered positive. Recommendations is denoted by *Recom*, evaluation is denoted by *eva*, neighbouring node is denoted by *g*, and *th* is the total quantity of neighbours.
Equation (11c) shows the formulation of the compile tips to decide the have confidence value for the trustee. Here, ∑ is used for the entire of all recommendations acquired from the close by nodes.

The following equation shows that nearby node is indicated by *g* and the suggestion as indicated by

$$\text{Recom}^{\text{check}}[a \rightarrow b] \quad (11a)$$

$$\text{Recom}_{c \rightarrow gth}^{eva} : [u_{g1 \rightarrow c} + u_{g2 \rightarrow c} + \text{------} + u_{gn \rightarrow c}] \quad (11b)$$

$$\text{Recom}_{c \rightarrow gth}^{form} = \sum_{0.0}^{1.0} (u_{g1 \rightarrow b} + u_{g2 \rightarrow b} + \text{------} + u_{gn \rightarrow b)}) \quad (11c)$$

When database is upgraded, Equation 12 compares the collected recommendation creation to the updated confidence score. *D* is the database, *T* is the value of trust, *b* is the trustee, *g* stand for neighboring nodes, and *th* is the quantity of total neighboring nodes. In Equation 13, trustee b is assigned the time-based certificate based on the developed trust value is represented by his TB^{cert}. After formulation step the determination of equation (14) is modified and is depend on a threshold. Delegator *a* assiststrustee *b* and start communication from adjacent hubs in case the final trust value is more prominent than or break even with to a foreordained threshold.

$$D^{update} : \text{Recom}_{u \rightarrow gth}^{form} = T_{a \rightarrow b}^{update} \quad (12)$$

$$TB^{cert} : \text{assign}(a \rightarrow b) \quad (13)$$

$$\text{if } (T_{a \rightarrow b}^{update} >= \text{threshold}) \quad (14)$$

IV. EXPERIMENTAL ASSESSMENT

This section compares the proposed mechanism's experimental results with the most recent trust management techniques, specifically GFMS [19], ETES [17],and SMTE [18]. Table II provides specifics on the experimental setting. Java was used for interaction, data was gathered using the Contiki Cooja simulator, and virtual machines were made available as a platform for simulations. The simulation aims to assess how well the recommended attack resistance and quality-of-service (QoS) strategy works. The proposed system's quality of service is evaluated by carefully examining the overhead ratio, latency, delivery ratio and reaction time of two nodes with other nodes.
The quality of service (QoS) of the recommended method is carefully assessed in terms of latency, overhead ratio, delivery ratio and response time of two interacting nodes resilience against different types of possible IoT attacks. We evaluated the system against a variety of IoT attacks, including whitewashing, self-promotion, and bad and positive mouthing, in order to verify its functionality from all aspects and gauge its resilience against these threats.

Based on the delivery ratio simulation results, the trust agent with the aid of 100 nodes continues to preserve a higher delivery ratio towards interacting nodes. The proposed system have to make certain that, even with 800 nodes, there is a steady go with the flow of data.

The simulation result for latency is displayed in Figure 3, demonstrating the effectiveness of the recommended strategy.

LightTrust functions better in networks with greater diversity because the trust agent's latency rises initially and falls over time.

Table 1: The simulation's constraints.

Constraints	Values
Simulator used	Contiki OS 2.7, Cooja OS
Nature of Deployment	Casual Position
Region use for Modeling	100(m) x 100(m)
Kind of Nodes	Sky Mote Type #sky 1
Number of Nodes used	100
Malicious Node	1:10
Rate of Transmission	4 Mbps
Wireless Protocol	IEEE 802.11

Time of Simulation	Varying between 10–100 (Minutes)
Delivery Rate	20 Packets per 60 Seconds

Trust agent is one of the critical component of the proposed trust management, which serves the edge nodes that is essential for defining the quality of service. To assess quality of service of the trust agent's, we looked at three important factors: overhead, latency, and delivery ratio. The delivery ratio was simulated with a range of nodes from 0 to 100, yielding the result shown in Figure 1. At 100 nodes, higher delivery rate to interacting nodes is handled by the trust agent, according to the delivery ratio simulation result. An important feature of the proposed system is that it keeps the data flow steady even when there are 100 nodes. The latency simulation results are shown in Figure 8, which illustrates the effectiveness of the recommended approach.

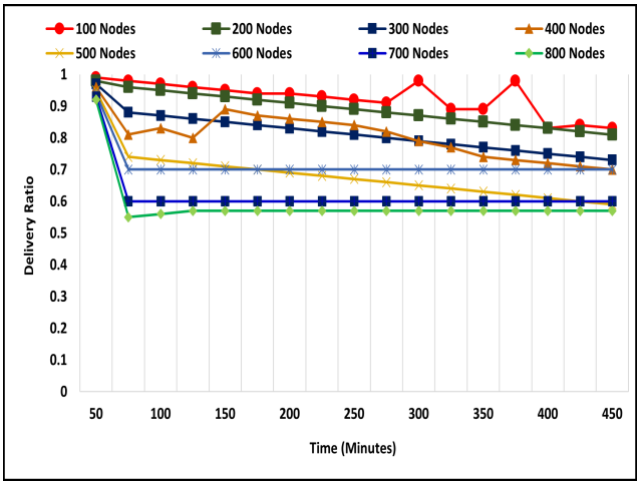

Figure 1. Evaluation of different nodes according to delivery ratio.

The results of the simulation, shown in Figure 2, demonstrate how well the suggested approach works in reducing l atency.

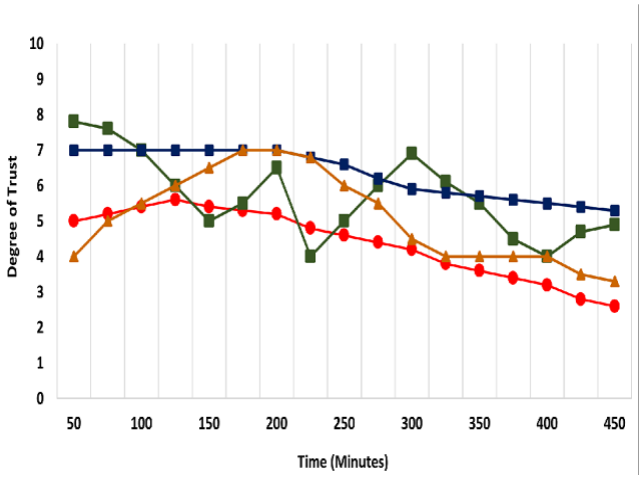

Figure 2. Latency ratio of the trustworthy agent.

Due to trust agent's initial rise and subsequent decrease in latency, the suggested methodology works better as the network becomes more heterogeneous. We calculated the overhead ratio in order to quantify the computational load and assess the functionality and utility of the trust agent. The simulation's output for the overhead ratio for nodes between 50 and 100 is shown in Figure 3.The overhead ratio for the recommended method rises from 200 to 375 for 100 nodes. The suggested approach is working well, even though the overhead tends to decrease and hits 275 as a result.

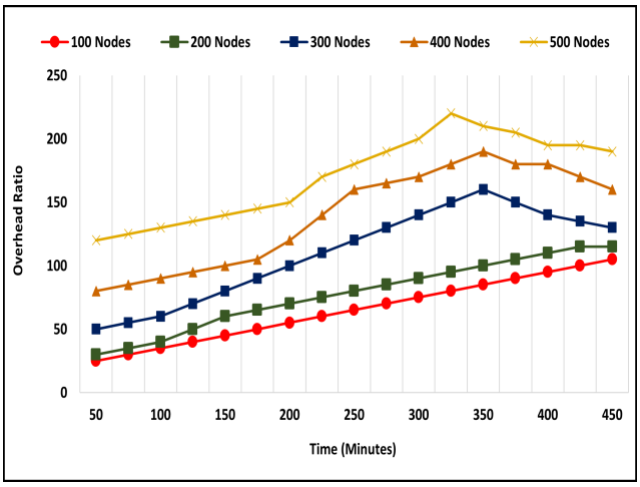

Figure 3. In terms of quality of service, the trust agent's overhead ratio is assessed

The recommended technique's robustness and resilience to different attacks were validated by evaluating its efficacy against possible attacks, such as good and bad mouthing, among others. Fifty of the 100 nodes that comprise the simulation environment are malicious or compromised. The simulation lasts between ten and one hundred minutes, and the default value for the trust limit is five. It can range from zero to ten.
Initially, nodes lean to trust hostile and compromised nodes. Good mouthing assault simulation output shown in Figure 4 demonstrates how the recommended approach first establishes a baseline of trust that gradually increases. After 150 minutes, the recommended approach recognizes the attacks successfully and begins to award smaller degrees, which eventually fall to 3.

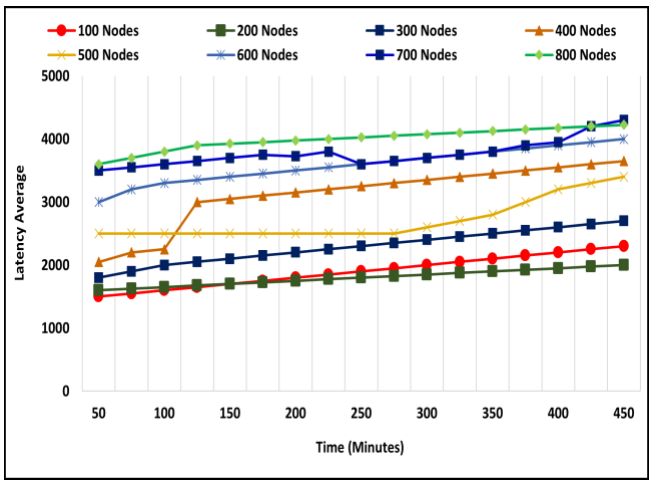

Figure 4. Performance of the presented method against goodmouthing

On the other hand, while older methods also detect attacks successfully, it takes longer to establish a lower degree of confidence

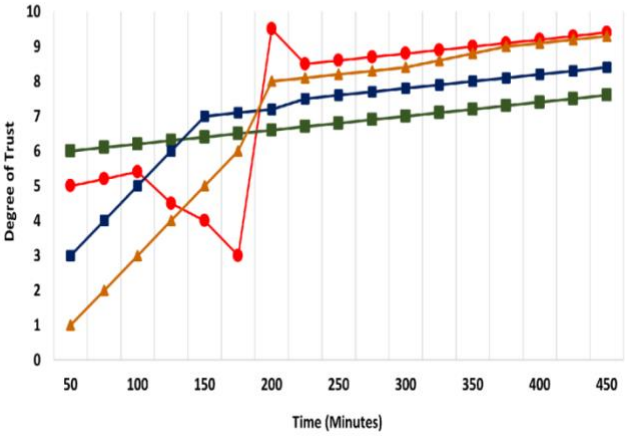

Figure 5.The simulation's result about the attack of bad mouthing.

Furthermore, we tested this performance in a different way: badmouthing, and simulated a situation as described in this section. Lower trust ratings are given to reliability-enhancing nodes as a means to de-highlight higher trust. Figure 5 shows the outcome of a simulation of a bad-mouthing attack. It shows how the proposed procedure quickly progresses from a lower initial degree of trust (> 3) to a higher one (> 8). It illustrates exactly what to look for when someone attempts to slander you.

Further, we have assessed the effectiveness of the strategy in resisting the whitewashing process. The simulation results in Figure 6 show that the default trust degree of the recommended mechanism is 5, which is lower than other trust degrees. The nodes' trust level starts to decrease after 25 minutes, indicating that the attacks were correctly detected.

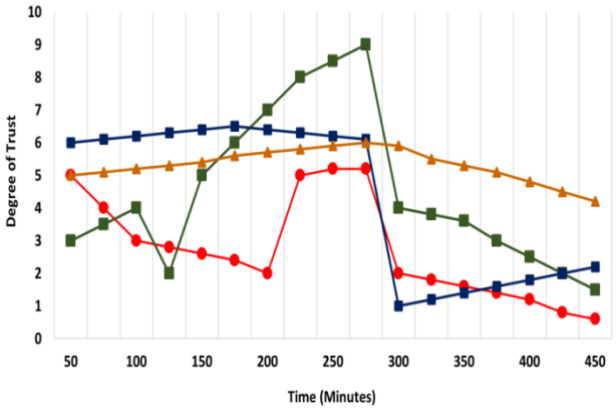

Figure 6.An evaluation of the suggested defense against whitewashing attacks

For additional proof of the suggested mechanism's efficacy, we evaluated its performance against self-promotional attacks. Figure 7 shows the outcomes of the simulation. It illustrates how the proposed method gives the network's participating nodes a higher trust degree at first, i.e., > 8, which could subsequently drop to > 4. However, it stays at that level, prohibiting nodes from increasing their level of trust again.

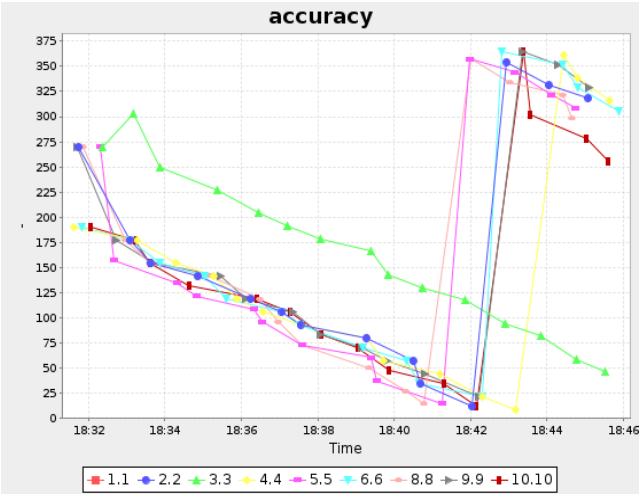

Figure 7. Divergence in Accuracy

Table 2. Assessment of result

Original LightTrust		Proposed Method	
Constraints	**Values**	**Constraints**	**Values**
Number of Nodes	800	Number of Nodes	100
Variation in time of Simulation	Between 50–500 (Minutes)	Variation in time of Simulation	Between 10–100 (Minutes)
Delivery Rate	8 Packets per 30 Seconds	Delivery Rate	20 Packets per 60 Seconds
Malicious Node	20:100	Malicious Node	1:10
Degree of Trust	> 7	Degree of Trust	> 8
Accuracy	Not considered	Accuracy	>9
Reliability	Not considered	Reliability	>9

V. Conclusion and Future Work

Trust is a critical component of the security and privacy of Industrial Internet of Things (IIoT) systems—where systems are interconnected, and "trust" among these systems, applications and users is necessary. Trust management plays a crucial role in ensuring secure communication by incorporating factors such as authentication, access control, data confidentiality, and behavioral reliability. The IIoT environments are composed of diverse and resource-limited devices; therefore, it is necessary to ensure reliable interactions between entities of the network to ensure trust.

IIoT networks use different communication technologies like Bluetooth, Zigbee, Wi-Fi, and cellular networks, along with suitable security measures, for accurate and reliable communication. These mechanisms guarantee the integrity, confidentiality and availability of the data in the exchange of information. Moreover, periodic calibration of industrial devices and accuracy of sensor readings are important to ensure the reliability of the system. Therefore, enabling secure and trusted data transfer between the interdependent nodes is a key prerequisite in the successful operation of IIoT systems.

The proposed extended lightweight trust management system overcomes these issues by integrating direct observations with recommendations-based trust assessment and adaptive trust. This allows the framework to detect malicious nodes accurately, resist the trust manipulation attacks, and ensure reliable communication while preserving lightweight properties essential for resource constrained edge environment.

The scalability and robustness of trust management systems have to be enhanced in the future by incorporating cutting-edge technologies, including blockchain, federated learning, and artificial intelligence. Decentralized and tamper-resistant trust records can be achieved via blockchain-based solutions, and data can be kept private while conducting collaborative trust assessment via federated learning. Furthermore, machine learning techniques could be further used to improve the malicious node detection and trust prediction. Along with increasing user awareness and promoting security best practices to further enhance the security posture of IIoT ecosystems. Wherever Times is specified, Times Roman or Times New Roman may be used. If neither is available on your word processor, please use the font closest in appearance to Times. Avoid using bit-mapped fonts if possible. True-Type 1 or Open Type fonts are preferred. Please embed symbol fonts, as well, for math, etc.

Acknowledgment

We would like to thank every entity and person who helped in completing this research. Special thanks are due to the faculty members and colleagues for their helpful suggestions, constructive criticism, and technical guidance that greatly enhanced this work.

References

[1] H. Li, L. Ge, and L. Tian, "Survey: Federated learning data security and privacy-preserving in edge-Internet of Things," Artificial Intelligence Review, vol. 57, no. 5, pp. 1–32, 2024.

[2] S. R. Alotaibi and M. A. Khan, "Security challenges in industrial Internet of Things: A comprehensive survey," IEEE Internet of Things Journal, vol. 11, no. 3, pp. 2105–2120, 2024.

[3] Y. Wang, X. Li, and J. Chen, "Trust management in industrial IoT: A multidimensional perspective," Future Generation Computer Systems, vol. 150, pp. 210–225, 2024.

[4] N. Kumar and P. Singh, "Lightweight IoT communication security mechanisms: A review," Computer Networks, vol. 236, pp. 110–123, 2024.

[5] V. Padmavathi and R. Saminathan, "Federated edge intelligence for secure IoT systems," Scientific Reports, vol. 15, 2025.

[6] A. Samanta and T. G. Nguyen, "Edge computing for industrial IoT: Resource optimization and latency reduction," IEEE Access, vol. 12, pp. 118900–118920, 2024.

[7] X. Zhang et al., "Trust-based security in IIoT edge networks," IEEE Internet of Things Journal, vol. 11, no. 7, pp. 12011–12025, 2024.

[8] S. Wang and Y. Zhao, "Security and trust frameworks for IoT systems," Future Generation Computer Systems, vol. 152, pp. 88–102, 2024.

[9] J. Kim and D. Park, "Secure trust establishment in IoT networks," IEEE Access, vol. 12, pp. 99800–99815, 2024.

[10] M. Ahmed et al., "Reliable trust evaluation in smart IoT systems," Computer Networks, vol. 240, pp. 109–125, 2024.

[11] X. Zhang et al., "Centralized vs decentralized trust management in IoT," IEEE Communications Surveys & Tutorials, vol. 26, no. 1, pp. 300–320, 2024.

[12] L. Chen and K. Liu, "Decentralized trust frameworks for IIoT systems," Ad Hoc Networks, vol. 150, pp. 103–118, 2024.

[13] A. Samanta and T. G. Nguyen, "LightTrust: Lightweight trust management in IIoT," IEEE Access, vol. 12, 2024.

[14] R. Sharma and P. Gupta, "Trust fundamentals in industrial IoT," IEEE IoT Journal, vol. 11, no. 6, pp. 9800–9815, 2024.

[15] Y. Wang et al., "Security and trust in industrial systems," Future Generation Computer Systems, vol. 155, pp. 120–135, 2024.

[16] N. Kumar and P. Singh, "Authentication and cryptography in IoT security," Computer Networks, vol. 236, pp. 110–123, 2024.

[17] Y. Zhang et al., "ETES-based trust evaluation using Dempster–Shafer theory," IEEE IoT Journal, vol. 11, no. 8, pp. 14235–14248, 2024.

[18] M. Khan and S. Ali, "Subjective trust models in IoT systems," Sensors, vol. 24, no. 10, 2024.

[19] R. Gupta et al., "Energy-efficient trust models for WSNs," Ad Hoc Networks, vol. 145, 2024.
[20] L. Zhang et al., "Trust management in delay-tolerant networks," IEEE IoT Journal, vol. 11, no. 9, 2024.
[21] S. Ali and M. Rehman, "QoS-based social trust in IoT," Computer Networks, vol. 234, 2024.
[22] J. Kim et al., "ScaleTM-IoT scalable trust framework," IEEE Access, vol. 12, 2024.
[23] A. Singh et al., "Sub Model-IoT trust estimation model," Sensors, vol. 24, 2024.
[24] H. Zhao et al., "Dynamic trust management in IoT CoI systems," Future Internet, vol. 16, 2024.
[25] S. Malik and A. Hussain, "Lightweight trust systems for IoT edge networks," IEEE Access, vol. 12, 2024.
[26] X. Zhang et al., "AI-driven trust prediction in industrial IoT," Computers & Electrical Engineering, vol. 122, 2025.
[27] Qamar, R., Zardari, B. A., Arain, A. A., Burdi, A., Kanwar, K., & Memon, E. F. A. A CONVOLUTIONAL NEURAL NETWORK-BASED MALWARE ANALYSIS, INTRUSION DETECTION, AND PREVENTION , University of Sindh Journal of Information and Communication Technology. Vol.6 Issue-4, 2022, https://sujo.usindh.edu.pk/index.php/USJICT/article/view/5695/4253
[28] Fatima, S., & Naz, L. F. Role of IoT in protecting wearable gadgets. University of Sindh Journal of Information and Communication Technology. Vol.5 Issue-4, 2021, https://sujo.usindh.edu.pk/index.php/USJICT/article/view/3883/3062