Trust Management for the LPWAN devices in a Smart City

Saifullah Tanvir, Zeeshan Ali Khan
NUCES, Lahore Pakistan
* Correspondence: {saifullah.tanvir, zeeshanali.khan}@nu.edu.pk

Citation | Khan. Z. A, Tanvir. S, "Trust Management for the LPWAN devices in a Smart City", IJIST, Vol. 07 Issue. 01 pp 392-410, Feb 2025
DOI | https://doi.org/10.33411/ijist/202571392410
Received| Jan 21, 2025 **Revised|** Feb 16, 2025 **Accepted|** Feb 20, 2025 **Published|** Feb 21, 2025.

The Internet of Things (IoT) is used in several domains like health care, transportation, military, banking, and many more. These applications can lead to the realization of a smart city application. Recently, Low Power Wide Area Networks (LPWAN) have been getting attention to implement various IoT applications. However, LPWAN devices are deployed in an environment where they can face malicious cyber-attacks leading to compromised data. To make successful network communication, security is an important factor that must be taken into consideration. Previously, many solutions involving sophisticated data encryption and machine learning techniques have been proposed for this purpose. However, they require processing power which is mostly not available in the LPWAN devices. Here, we can apply lightweight trust management techniques to find the reliability of a node. In this article, we propose a trust management framework for securing LPWAN-based Smart City applications. Multiple Smart City case studies are considered for evaluating the proposed technique and results show better intruder detection.

Keywords: Smart cities; Sustainable development; Infrastructure; Energy efficiency; Security; Urban Planning

Introduction:

A smart city leverages IT-based services and applications to enhance the quality of life for its residents. This includes the implementation of emerging technologies such as LPWAN-based IoT networks, enabling efficient and innovative urban solutions [1]. In [2], a smart city is defined as a technologically advanced system designed to enhance the quality of life for its residents through innovative and efficient solutions. While [3] considers a smart city as a way to provide sustainable and efficient services to the residents. Furthermore, [4] defines a smart city as "when investments in human and social capital and traditional (transport) and modern (ICT) communication infrastructure fuel sustainable economic growth and a high quality of life, with a wise management of natural resources, through participatory governance". The European Union is supporting multiple projects under its umbrella to implement the concept of smart cities, focusing on sustainability, digital innovation, and efficient urban services [5]. Therefore, it is crucial to implement this concept by leveraging advancements in communication technology. Due to the nature of these devices, they are vulnerable to security attacks and they may need to be protected in an energy-efficient way. Trust management provides a way to secure these devices by considering these constraints.

LPWAN devices collectively form a system that enables sensing, control, and monitoring activities. These devices can collect and share data with other nodes, process it locally, and send it to a base station or a central hub. Factors including memory, speed, and processing play a crucial role in determining the architecture most suitable for the network. Additionally, LPWAN devices have unique identifiers such as self-configuring abilities, allowing a large number of devices to operate together seamlessly to provide specific services [6].

The primary objective of this paper is to explore the use of Low Power Wide Area Networks (LPWAN) for smart city applications. Additionally, LPWAN is gaining significant attention due to its ability to provide cost-effective communication for low-power distributed IoT devices. It is designed for long-range communication while maintaining low data rates. The power consumption of these devices depends on the amount of data transmitted. The lower the data rate, the farther the data can be transmitted. Sigfox and LoRa are among the pioneering technologies in this field [7]. Sigfox offers an end-to-end LPWAN connectivity solution. It uses data transmission while minimizing interference resulting in high receiver efficiency, which results in a throughput of 100bs. Initially, Sigfox supported uplink communication, having limitations on both the number and size of messages. The uplink is restricted to 120–140 messages of 12 bytes per day, while the downlink allows only 4 transmissions of 8 bytes per day [7]. Whereas, LoRa is a physical layer innovation that modulates the signals in the SUB-GHz ISM band utilizing a restrictive spread range procedure created and popularized by Smetech Corporation. It supports multiple spreading factors (ranging from 7 to 12), allowing for a tradeoff between range and data rate. Higher spreading factors enable longer communication ranges at the expense of lower data rates, while lower spreading factors provide higher data rates with reduced range [7].

The machine learning algorithms for the security of LPWAN devices require a massive amount of data to produce structured results and on the other hand, these devices collect and exchange lots of data. Thus, executing machine learning algorithms on these devices is not feasible due to their low voltage power, memory, and processing power, which is only sufficient enough to process its data. The frequency of attacks on such devices is increasing rapidly due to their affordability and speed. Malicious attacks on sensitive sectors like banking and healthcare can result in significant losses. Furthermore, if a company or brand fails to provide a secure network to its clients, it risks losing trust and can lead to significant damage to the company. For example, if a patient is connected to a device to monitor their heartbeat and some malicious attack can change the data, the consequences of this attack can even cause

death or serious damage to the patient by not making the right decision based on that information. Similarly, in agriculture, if IoT devices are deployed to monitor crop conditions and a cyberattack manipulates the data, it could lead to poor decision-making regarding harvest timing, ultimately causing financial losses for the company.

This study aims to explore various trust-based security solutions that can effectively minimize energy consumption while ensuring robust protection against cyber threats for smart city applications. We have examined multiple case studies demonstrating the implementation of LPWAN devices in smart cities, which are detailed in the following sections.

Following are some of the examples of cyber-attacks in the LPWAN networks for Smart Cities:

- Sybil attack: The attacking node generates false information to conceal its identity. The malicious node creates multiple identities and manipulates the network to disrupt its operations [8].
- Crude attack: Node creates false information about their trust value. It is that type of attack in which the node generates opposite results to its sensing outcomes [9].
- Denial of Service attack: Malicious node prevents data forwarding. It's a type of attack in which the attackers try to shut down or block the network by flooding the targeted traffic [10].
- Black Hole attack: Node considers them as good packet forwarders but drops the packets as received. It also considers itself as the shortest path to the destination in the network. So, the source can send packets to it [11].
- Routing Attack: The attack takes place during message routing and specifically targets the network layer [12].
- Bad-Mouthing: The attacking node makes incorrect information about the rest of the nodes to reduce their reputation [13].
- On/Off Attack: Nodes keep changing their condition, as good or bad nodes to disturb the trust scheme [14].
- Good-Mouthing: Malicious nodes give positive feedback to quickly gain a high reputation [13].
- Opportunistic Attack: This is the type of attack in which, the attacker waits for a vulnerability for an opportunity to perform an attack [15].
- Selective Forwarding Attack: A malicious node drops selective packets and forwards the rest of the packets [16].
- Sinkhole Attack: it is a type of attack in which the malicious node gives itself a low rank so that its neighbors select it as the parent node. The malicious node advertises itself by creating fake routing updates [17].
- Worm Hole Attack: This is a type of attack where a node listens to the network without making any changes in the network. According to Science Direct [18], in this type of assault, two bad sensor hubs burrow control and information bundles between one another, fully intent on making an easy route for themselves in the remote sensor organization. These two scheming hubs will probably build the likelihood of it being chosen as a functioning way.
- Grey Hole Attack: The attacking node acts as a central controller to gain information from all nodes in the network. Moreover, it switches from becoming a normal node to a sinkhole node. It becomes typical to decide whether it is a normal or malicious node [19].
- Eavesdropping: To listen to the conversation in a network, secretly. The attacker takes advantage of a network that is not secured, by accessing the data communicated

between the nodes [20].

- Repudiation Attack: It is a type of attack in which the system does not properly track the logs of events. It becomes hard to detect malicious attacks as after the attack the data generated on the log files can be invalid or misleading [21].
- Ballot-Stuffing Attack: It is a type of attack in which the attackers increase the reputation of other nodes by increasing their recommendation value [22].

Literature Review:

This section explores various techniques proposed for detecting malicious nodes in different IoT networks. These techniques have different parameters, some are LPWAN enabled and some are not. Analyzing these approaches will aid in developing a more effective solution for LPWAN-enabled networks. Gao, Kanhere, Jha, and Hu [23] proposed a model in which an encryption key is generated using various signal processing techniques and real-time received signal strength indicators to enhance the key generation rate. His approach is not suitable for the current study as it deals with the device's physical layer to generate the keys. Our focus is on software-based security techniques for the LPWAN networks. Furthermore, this technique requires high computational power, which might not be available in the LPWAN devices.

Ahmad, Yau, Ling, and Keoh [24] developed a hybrid framework that includes both centralized and distributed approaches to cater to dynamic schemes and heterogeneous schemes. In a centralized approach, the trust values are managed based on the node ID, location, and residual energy on the central entity. In the distributed scheme, the same process is applied to individual nodes. However, since this framework utilizes both centralized and distributed approaches, it may demand significant processing power to gather network-wide information, making it inefficient for low-power devices. Additionally, a single failure in the central hub could disrupt the entire network [24]. Therefore, we can use a similar procedure by customizing it for LPWAN Devices.

Ribeiro, Filho, and Ramos [25] presented some features to secure the end device and the network. This paper proposes a dual-layer encryption approach, with one layer securing the application layer and the other protecting the network layer. This encryption technique is based on an advanced encryption standard algorithm. It operates as a distributed network, where end devices must authenticate before joining the LoRaWAN network. However, authentication through Activation by Personalization (ABP) has a drawback: the encryption key remains the same throughout the lifetime of the end device, making it vulnerable to security risks. Moreover, it requires high processing power for resource-constrained LoRaWAN devices.

Awan, Din, Zareei, Talha, Guizani, and Jadoon [26] developed the HoliTrust model, which is a mix of different centralized authorities that registers trust by gathering trust from various concentrated specialists that incorporate community's server, domain server, and trust servers. Community servers are restricted from communicating across different domains. HoliTrust framework provides multilevel security and each community has its server to calculate the trust value. But it generally targets generic IoT networks. Conversely, our focus is solely on LPWAN-based devices. Awan, Din, Almogren, Guizani, Altameem, and Jadoon [27] provided a trust management system proposed for IoT, which is known as vigorous cross-area trust management (RobustTrust). The system consists of multiple components designed to establish trust, providing nodes with resilience against various types of attacks. The trust model is occasionally driven, which implies that a node possibly assesses trust when an occasion happens between two nodes. Because of occasional trust calculation, a bad node may become part of the network. Furthermore, to use this approach for LPWAN, we have to modify it accordingly. Harsányi, Kiss, and Szirányi [28] presented a model to detect wormhole attacks on wireless sensor networks. However, it is a memory-consuming process, which is

not very suitable for low-power devices such as LPWAN.

Khan and Herrmann [22] applied a trust framework that is applied on an RPL-based IoT network, where three factors were used to determine the trust value of a node. Those factors are belief, disbelief, and uncertainty all these values are between 0 to 1 and their total sum must always be added to 1. These factors can also be used for trust calculation in an LPWN-based network. Ye, Wen, Liu, Song, and Fu [29] worked on a model in which multiple trust factors are included. The main focus of this model is to secure routing and information. The model relies on multiple trust factors, which may lead to additional memory and processing consumption, posing a challenge for IoT devices with limited resources.

Alsaedi, Hashim, Sali, and Rokhani [30]] demonstrated a framework to avoid Sybil's attack, which comprises multi-level detection to eliminate Sybil attacks using the clustering technique. The clustering technique is used to reduce the communication overhead and energy consumption. This model only works for Sybil attacks, which is not an optimal solution to protect from many malicious attacks. Y. Kim, Kim and Park [31] worked on a model, where the gateway is linked with the mobile edge computing server (MEC), which helps in processing a large amount of data. The MEC server classifies nodes as trustworthy or untrustworthy within the network using a logistic regression algorithm. This approach is not preferred, as it requires the addition of an MEC server, which becomes costly, and executing a logistic regression algorithm will also need high processing power.

Hellaoui, Bouabdallah, and Koudil [32] proposed a network of heterogeneous objects where only authenticated nodes are allowed to post messages. However, an adversary, lacking proper authentication, may attempt to post arbitrary messages in the network, leading to unnecessary resource consumption. Every node concludes locally to verify the message or not relying upon the trust level that it partners to the message sender (neighbor that gave or transferred the message). In this model, authentication is performed only when necessary, making it energy-efficient. This is a crucial factor that could be beneficial for our security model. But this approach only targets general IoT networks and to use it for LPWAN networks, it needs to be modified accordingly.

Wang, Wu, Chen, Ye, Zhang, and Zou [33] presents a model where the decision about a node is made using the Markov decision process. After calculating a node's trust value, a policy is assigned based on its trust level. The policy assignment decision is made using the Markov decision process. This type of model is preferable, and we can further optimize it by reducing the number of processes in the Markov decision process to minimize processing and memory consumption. Chen, Guo, and Bao [34] worked on an adaptive IoT trust mechanism, we have four main factors upon which trust value is generated. A Bayesian Framework is being used to calculate the direct trust value, which has a high communication overhead and requires high processing power.

Fayaz, Mehmood, Khan, Abbas, and Gwak, [35] use a reputation-based mechanism for finding a selfish node in an Adhoc Network to quickly identify, namely counteracting selfish nodes using a reputation-based system technique (CSNRS). Because of its lightweight property, it is a good candidate for LPWAN networks and we use this idea to compare with our proposed solution. Herrmann [36] proposed a model where the trust value is first calculated, and then policies are assigned to nodes based on their trust levels. We prefer this model because it dynamically assigns policies according to trust values, which helps reduce power consumption by performing authentication only when necessary.

Comparing different security techniques can help identify the most efficient approach for optimal performance. However, some techniques may be difficult to implement due to the limited processing power of LPWAN devices. Most of the work related to LPWAN network security presented is the Advance Encryption Standard (AES-128) protocol for message encryption. Some security techniques presented are more hardware-oriented and

some use machine learning algorithms for securing the network which requires high processing power. This paper aims to propose a trust management-based solution that efficiently addresses processing power, memory, and energy constraints in LPWAN-enabled devices, building upon the relevant literature discussed above. A comparative Analysis of these techniques is also presented in Table 1.

Objectives and Novelty:

LPWAN-enabled devices can send data packets for long-range and have long battery life. One such network is depicted in Figure 1, where multiple LPWAN devices are connected to a cloud-based network. For such a network, the security techniques discussed in the literature review are not well-suited, as most require high processing power, and some demand significant memory capacity, making them impractical for resource-constrained environments. As a result, these techniques can lead to rapid battery drainage, ultimately reducing the overall lifespan of the device's battery. Due to the limited literature available on trust management for LPWAN, this paper aims to implement algorithms within the LPWAN network to detect suspicious node behavior. These networks have a limited amount of computing power and memory storage; therefore, they cannot execute frameworks like statistical data analysis, machine learning, and other algorithms that encrypt/decrypt large keys for security, utilizing high processing power. Therefore, we preferred a lightweight trust management mechanism tailored to the LPWAN protocol.

Research Methodology:

In this section, we discussed the research methodology of the proposed research on trust management in LPWAN networks. First, we shall discuss the need for such a work based on the literature review. Then, we shall detail the proposed mechanism.

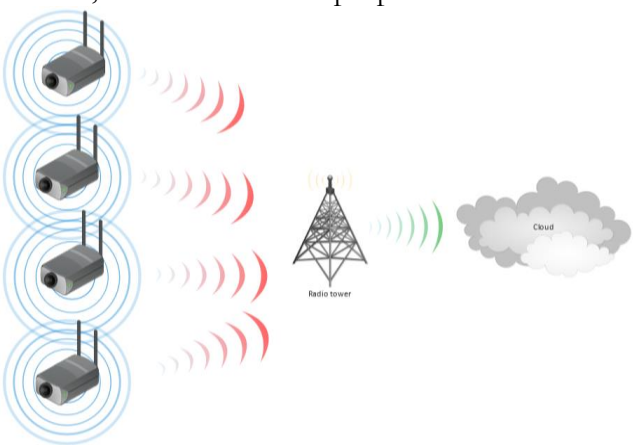

Figure 1. An example LPWAN network scenario

The next sub-sections detail the proposed algorithm and corresponding use cases for the Smart City application.

Trust Management for LPWAN-based Networks:

In LPWAN, the network follows a centralized architecture consisting of end devices, gateways, and the cloud, where data transmission and management are coordinated through these interconnected components. In Figure 2, the LPWAN network consists of four components: the green node represents an end device, the red node indicates a malicious device, the blue node serves as a gateway, and the final component is the cloud, which processes and manages data. The end nodes send data to the gateway, and from the gateway, it is transferred to the cloud. The cloud is the central hub, where all the data is stored. End nodes cannot communicate directly. The communication between the end nodes and gateway is bi-directional. We can also have more than one gateway in our LPWAN network.

The green and red nodes within range of each other can monitor the packets of their neighboring nodes and assign trust values based on three factors: Belief, Disbelief, and

Uncertainty. The red node is malicious so its neighboring node is assigned a value based on the number of packets forwarded and dropped. In the final step, each node transmits the trust value of its neighboring nodes to the blue node, which functions as the gateway. The gateway then calculated the overall trust value. Since our network follows a centralized architecture, the gateway is responsible for making the final decision on whether a node is malicious or not. The gateway monitored malicious nodes by checking whether the disbelief value exceeds the assigned threshold. If it is greater than the threshold, the node is classified as malicious.

Table 1. Comparative Analysis of the Literature Review

Sr#	Reference	Centralized	Distributed	LPWAN Enabled	Research Contributions	Limitations/ Future Recommendations	Processing Power
1.	[22]	×	✓	×	Misbehavior Detection for bad-mouthing, self-promoting & Ballot-stuffing attacks	Need to work on the energy consumption of a node as well	Medium
2.	[23]	×	✓	×	Signal processing techniques to increase the key generation rate	High energy consumption	High
3.	[24]	✓	✓	×	Misbehavior Detection for blackhole, Sybil, wormhole, crude attacks, and routing attacks	High Heterogeneity & High Dynamicity	High
4.	[26]	✓	×	×	Misbehavior Detection for several attacks & Cross-domain trust management model	High Energy Consumption & High Communication overhead.	High
5.	[27]	×	✓	×	Misbehavior Detection for several attacks & Cross-domain trust management model	Memory consumption & No Lightweight cross-domain model	High
6	[25]	×	✓	✓	The exploitation of LPWAN Server	Need a mechanism to update the encryption key	Medium
7.	[28]	×	✓	×	Misbehavior Detection for wormhole attacks	High processing consumption	High
8.	[29]	×	✓	×	Malicious behavior detection for several attacks with a punishment factor	Multiple Trust factors	Medium

9.	[30]	✓	✓	×	Misbehavior Detection & Detects Sybil Attack	It only avoids Sybil's attack	Medium
10.	[31]	✓	×	✓	To make trusted connections for data transmission	Requires addition of mobile edge computing server to execute logistic regression algorithm	High
11.	[32]	×	✓	×	Misbehavior Detection for on/off attack & Energy Efficient	Need to work on avoidance of mouthing attack	Low
12.	[34]	×	✓	×	Misbehavior Detection for bad-mouthing & self-promoting attacks	High Energy Consumption & High communication	High
13.	[35]	✓	×	×	Reputation management for Adhoc Networks	Proposed for Adhoc Networks only	Low
14.	[36]	✓	×	×	Generic Trust Management Framework for any application	Evaluate the proposed mechanism for emerging technologies	Low

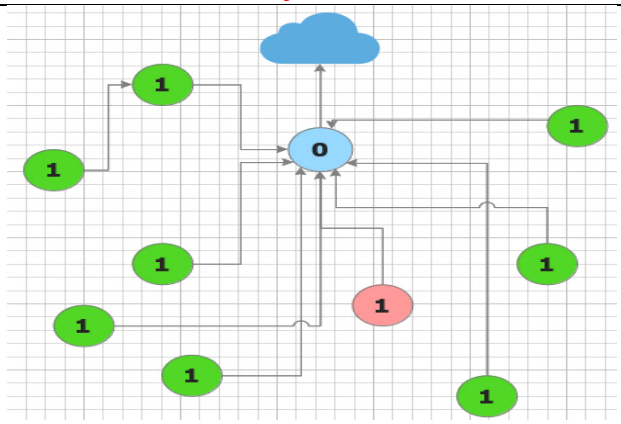

Figure 2. Trust Management in LPWAN Network

If the distance between two nodes is less than a specified threshold, they are considered neighbors. Both of these nodes can assign trust values to each other based on packets forwarded or dropped. Moreover, they utilize the following formula to calculate the trust value of their neighbor node:

$$\textbf{belief} = \frac{\mathbf{p}}{\mathbf{p}+\mathbf{n}+\mathbf{k}}$$

$$\textbf{disbelief} = \frac{\mathbf{n}}{\mathbf{p}+\mathbf{n}+\mathbf{k}}$$

$$\textbf{uncertainty} = \frac{\mathbf{k}}{\mathbf{p}+\mathbf{n}+\mathbf{k}}$$

Where "p" is a positive interaction, "n" is a negative interaction, and "k" is a constant as k=1. The value of these variables of the trust value vector [belief, disbelief, uncertainty] varies between 0 and 1, and their sum must be equal to 1. The above formula can be used by end devices to calculate the trust of their neighbor nodes.

Trust Algorithm Computation in an End Device:

Algorithm 1 operates on the end devices, where each device continuously monitors its neighboring devices. If any suspicious activity is detected, the trust value of the suspected malicious device is decreased by its neighbors. These trust values are then forwarded to the gateway for final evaluation and trust calculation.

Algorithm 1: Algorithm computing in the end device

```
Insert Trust (neighbor list, positive Interactions, negative Interactions)
{
  Do for all neighbors.
  {
    belief = positive Interactions / (positive Interactions + negative Interactions + 1)
    disbelief = negative Interactions / (positive Interactions + negative Interactions + 1)
    uncertainty = 1 / (positive Interactions + negative Interactions + 1)
    Assign trust value (belief, disbelief, uncertainty) to a neighboring node
  }
}
```

Trust Algorithm Computation in the Gateway

When the end devices send trust values of their neighbor nodes to the gateway, it calculates the overall trust value of end devices and then checks if that node is malicious or not. Algorithm 2 is based on combining trust values using the ⊕ operator, which is called the consensus operator [36].

The overall trust values are calculated as follows. The Vxy = [Bxy, Dxy, Uxy] is the trust value vector of a node Y stored in its neighbor node X and Vzy = [Bzy, Dzy, Uzy] is the

trust value vector of node Y stored in its other neighbor node Z. The combined overall trust value of node Y is Vxy ⊕ Vzy, which is calculated by the gateway, where 'K' is [(Uxy + Uzy – (Uxy * Uzy)]. The formulas for calculating the overall trust value are shown below [36]:

$$\text{belief} = \frac{(\text{Bxy} * \text{Uzy} + \text{Bzy} * \text{Uxy})}{\text{K}}$$

$$\text{disbelief} = \frac{(\text{Dxy} * \text{Uzy} + \text{Dzy} * \text{Uxy})}{\text{K}}$$

$$\text{uncertainty} = \frac{(\text{Uzy} * \text{Uxy})}{\text{K}}$$

Algorithm 2: Algorithm computing in the gateway

```
Calculate Trust Overall Trust
{
   Get the values of belief, disbelief, and uncertainty from the entire network.
   Combine trust values for every node
   For all nodes
        If Disbelief > Disbelief_Threshold
        {
              Set it as a malicious node.
        }
}
```

Implementation of Methodology:

As discussed in Algorithms 1 and 2, the gateway devices computed the trust value for all the devices using Trust management techniques. Thus, the network was secured using these lightweight techniques as defined in Section 3.2. of this paper. A glimpse of this technique is given in Figure 3, where a central node/gateway periodically receives the trust values from all the end devices and it combines those values using the Subjective Logic's consensus operator [36]. In the forthcoming sections, the limitations of this study are presented followed by some real-world case studies of the smart city.

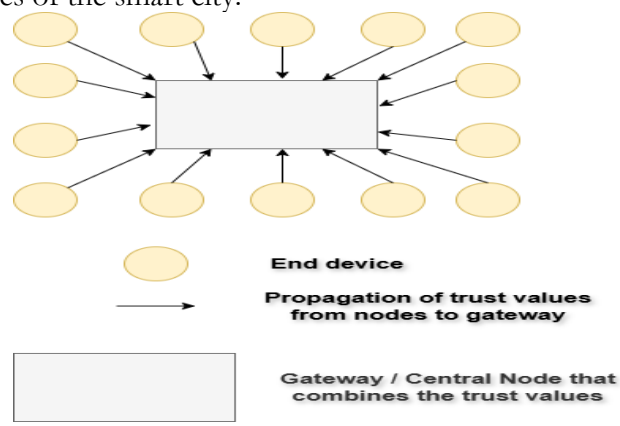

Figure 3. Block Diagram for implementation of trust management technique to secure LPWAN

Assumptions and Limitations:

Following are the assumptions made by our technique:

1 The network model is not formulated by the actual device but is assumed to have the same outcomes as provided by the actual devices for LPWAN.
2 The type of data packets being transferred are assumed to be similar. However, the different types of data packets can provide different results.
3 The approach can be applied to any form of data transfer protocol e.g., cellular technology, etc. However, LPWAN has been selected in this paper for evaluating trust management

systems as not much literature is available for LPWAN.

4 The data transfer rate is assumed to be linear and follows similar patterns in transfer from one end to another.

5 Noise/interference i.e., the loss of data packets is assumed to be "none", as the data packets sent from one end to the other are assumed to be efficient. Interference may lead to a loss in efficiency.

Case-Study for real-world applications:

In the following subsections, we detailed the use of the Trust management framework in the context of three use cases of a Smart City application. In the forthcoming section, we created several images (Figures 4-6) to illustrate these smart city applications using Microsoft Paint, by combining various online images. To validate these use cases, we used a C++-based simulation framework that evaluates the proposed security techniques for these applications.

Intruder Trespassing Secure Building:

In Figure 4, a network deployed around the border of a secure place is used to detect the presence of unwanted intruders. This system is used to detect humans around that specific area. The LPWAN-enabled end devices embedded with PIR sensor are deployed around the borders and some gateways, for the end devices to share data. If an end device is being attacked by a Denial-of-service attack by some attacker it pings that end device many times to send data to the gateway. During this process, there are chances that data packets might drop. There can be a black hole attack on the end device, where the attacker makes some packets dropped by the end device, and in the same way, a selective forwarding attack makes some packets drop and forward the rest.

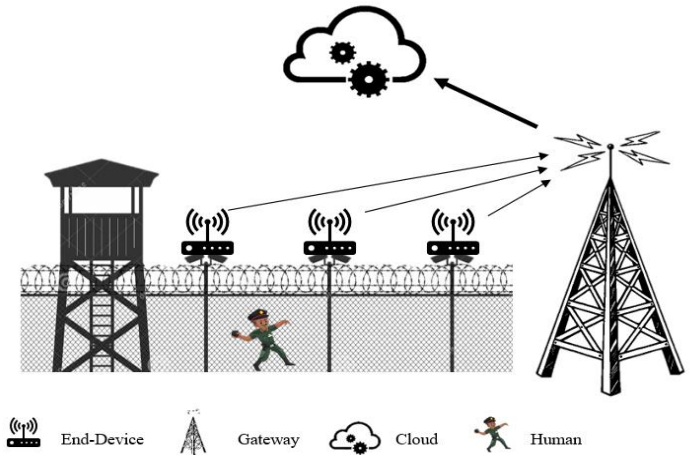

Figure 4. Case-Study Diagram (Intruder Trespassing Secure Building**)**

University Car Parking:

In another case in Figure 5, LPWAN-enabled devices were deployed in universities to detect unauthorized cars that cannot park their cars on university premises. We assumed that some end devices were affected by black hole attacks and to detect those end devices using the trust management techniques proposed in this paper. Any suspicious car that attempts to enter the university premises can be detected and instantly reported to the security department to avoid any inconvenience.

Traffic Monitoring:

Lastly, LPWAN-enabled devices were deployed in a city to detect cars that break traffic rules. Figure 6 shows more than one gateway, where several end devices send data to these gateways. We assumed some end devices suffered from selective forwarding attacks. Our proposed trust management technique aims to detect these affected nodes and any traffic violation can be instantaneously reported to the traffic department.

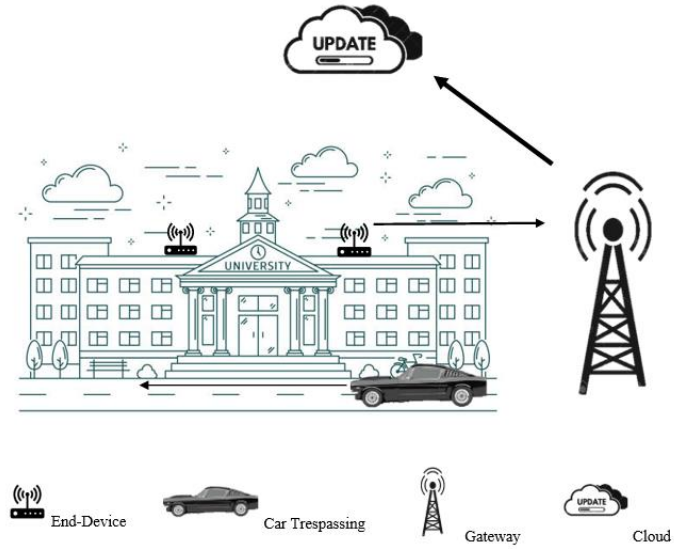

Figure 5. Case-Study Diagram (parking car)

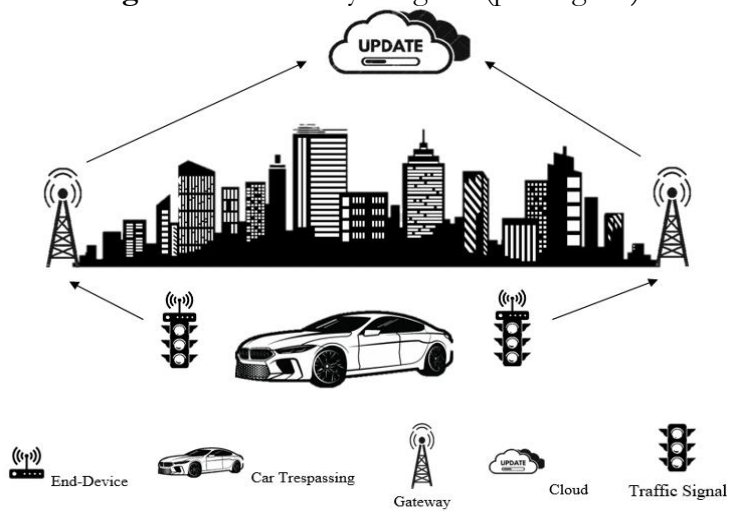

Figure 6. Case-Study Diagram (traffic monitoring)

Evaluation Parameters:

We evaluated our techniques by observing the number of malicious nodes detected and the number of malicious not detected. Moreover, we used a confusion matrix to find the accuracy rate and error rate. We employed the following properties in our confusion matrix: TN (True Negative): When there is no malicious node in the network and the algorithm also doesn't detect any malicious node, it is known as true negative. TP (True Positive): When there exists a malicious node in the network and the algorithm also detects the malicious node, it is known as true positive.

FN (False Negative): When there exists a malicious node in the network and the algorithm does not detect that malicious node, it is known as a false negative.

FP (False Positive): When there is no malicious node in the network and the algorithm detects a malicious node, it is known as a false positive.

The formula to calculate the accuracy rate is:

$$\text{Accuracy rate} = [(TP + TN) / \text{Total}]$$

The formula to calculate the error rate is:

$$\text{Error rate} = [(FP + FN) / \text{Total}]$$

Where 'Total' is (TN + FP) + (FN+TP) = (TN+FN) + (FP+TP).

Results: We experimented with our proposed TM methodology and the CSNRS technique for counteracting selfish nodes to evaluate their accuracy and error rates. We utilized this

approach to determine improvement in the detection of malicious nodes that are affected by black hole attacks, selective forwarding attacks, or denial-of-service attacks. Below we detail and analyze the simulation results.

Simulation Results:

This section provides comparisons between our Trust Management (TM) technique and the CSNRS technique that is explained in the literature review. For all the experiments, we assumed that some end devices have been subjected to cyber security attacks and they are intentionally dropping the packets. By using a random function, we dropped the number of packets of these malicious nodes to replicate that scenario in the simulation. Both of these techniques (i.e., TM and CSNRS) were implemented in these three case studies using a simulation program written in C++.

Results for Case Study-1 (Intruder Trespassing Secure Building):

In the first experiment, we considered a case for LPWAN-enabled devices embedded with a PIR sensor to detect the presence of humans around the national border of a country to secure a national defense system. For the simulations, the number of infected nodes is varied and the detection rate of malicious nodes is calculated using TM and CSNRS techniques. Changing the number of infected nodes gives us better comparisons of both techniques. The simulated network size is 350 end devices, having 15 gateways and 15-20 malicious nodes. It was also considered that each end device has a total of 100 packets that were forwarded to the gateway. The parameters for simulation are also listed in Table 2.

Every node assigns a trust value to its neighbor node based on its positive or negative activity. In the end, every node sent the trust value of its neighbor to the gateway, where it calculated its overall trust value. The number of positive (packets forwarded) and negative (packets dropped) interactions with other end devices in the network remain the same for both techniques in all experiments.

Table 2. Parameters for first case study (Intruder Trespassing Secure Building**)** experimentation

Parameters	Values
End Devices	350
Gate Way	15
Number of Malicious Nodes	5 - 20
Algorithms	TM, CSNRS

The position of the gateways and end devices remained the same and also the malicious nodes were the same for both of the malicious node detection techniques. The results of Case Study 1 are listed in Table 3, Table 4, and Figure 7. According to this information, the accuracy rate of counteracting selfish nodes using the reputation-based system technique was 0.99, and the error rate was 0.027. In the same way, the accuracy rate of our proposed technique was 0.99, and the error rate was 0.007. The reason our third bar reading shows the highest detection of malicious end devices is that these devices had more than ten neighbors. Thus, the more neighbors an end device has, the more accurate the results will be. This is due to the use of Subjective Logic [36], which classified the nodes as not only being good or bad but also categorized them as "uncertain" when sufficient data is not available.

Table 3. TM Confusion Matrix for case study 1 (Intruder Trespassing Secure Building**)**

TM Confusion Matrix	**Predicted**			
		Non-Malicious	**Malicious**	
Actual	**Non-Malicious**	[True Negative] 1350	[False Positive] 0	1350

	Malicious	[False Negative] 3	[True Positive] 47	50
		1353	47	1400

Table 4. CSNRS Confusion Matrix for Case Study 1 (Intruder Trespassing Secure Building**)**

CSNRS Confusion Matrix	**Predicted**			
Actual		**Non-Malicious**	**Malicious**	
	Non-Malicious	[True Negative] 1350	[False Positive] 0	1350
	Malicious	[False Negative] 10	[True Positive] 40	50
		1360	40	1400

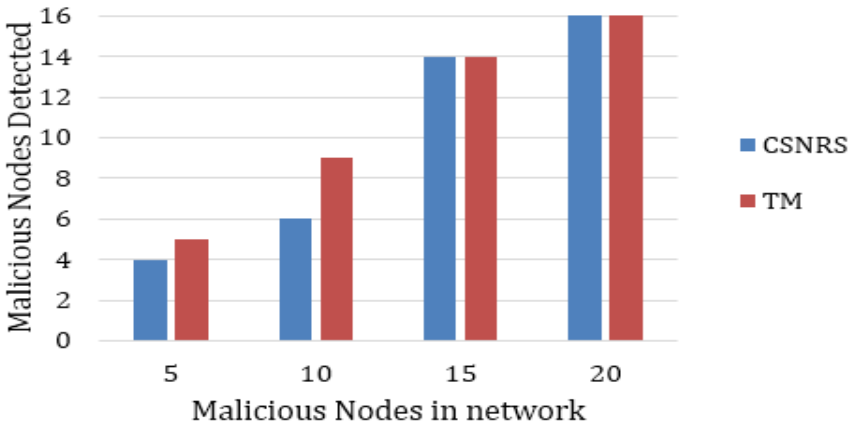

Figure 7. Comparative analysis for case study-1(Intruder Trespassing Secure Building**)**

Results for Case Study-2 (University Car Parking):

In our second experiment, end devices monitored the road traffic and found out the vehicle broke the traffic rules by not stopping before the limit line which is pavement marking on the road, when the traffic signal was solid red light. Moreover, there were some positions of end devices that were denser. Thus, we observed the impact on the network, where the end devices have many neighbors. Table 5 describes the simulation parameters for the second case study. The results for Case Study 2 are listed in Table 6, Table 7, and Figure 8. According to this information, for the case study of traffic signal vehicle monitoring, the accuracy rate of counteracting selfish nodes using a reputation-based system technique was 0.96, and the error rate was 0.03. In the same way, the accuracy rate of our proposed technique was 0.99, and the error rate was 0.009. This is due to the use of Subjective Logic, which classified the nodes as not only malicious or non-malicious, but also categorized them as "uncertain" when enough data is not available.

Table 5. Parameters for the second case study (University Car Parking) experimentation

Parameters	**Values**
End Devices	110
Gate Way	7
Number of Malicious Nodes	5 - 20

Table 6. TM Confusion Matrix for Case Study 2 (University Car Parking)

TM Confusion	**Predicted**

Matrix				
Actual		Non-Malicious	Malicious	
	Non-Malicious	[True Negative] 390	[False Positive] 0	390
	Malicious	[False Negative] 4	[True Positive] 46	50
		394	46	440

Table 7. CSNRS Confusion Matrix for Case Study 2 (University Car Parking)

CSNRS Confusion Matrix	Predicted			
Actual		Non-Malicious	Malicious	
	Non-Malicious	[True Negative] 390	[False Positive] 0	390
	Malicious	[False Negative] 14	[True Positive] 36	50
		404	36	440

Results for Case Study-3 (Traffic Monitoring):

In our last experiment, we considered a case where LPWAN devices were deployed on university premises to detect vehicles that were not registered to park or drive their vehicles in the university. In this case, we had only one gateway and fifteen end devices. Some of the nodes were compromised by malicious attacks. We then executed our proposed TM technique and the CSNRS technique for counteracting selfish nodes to detect these malicious nodes.

Table 8 describes the simulation parameters for the second case study. The results for Case Study 2 are listed in Table 9, Table 10, and Figure 9. According to this information, for a case study of university car parking, the accuracy rate of counteracting selfish nodes using a reputation-based system technique was 0.83, and the error rate was 0.16. In the same way, the accuracy rate of our proposed technique was 0.95, and the error rate was 0.05. As concluded above, the use of Subjective logic [36] helped us achieve better results than the CSNRS, as its mechanism used more sophisticated metrics.

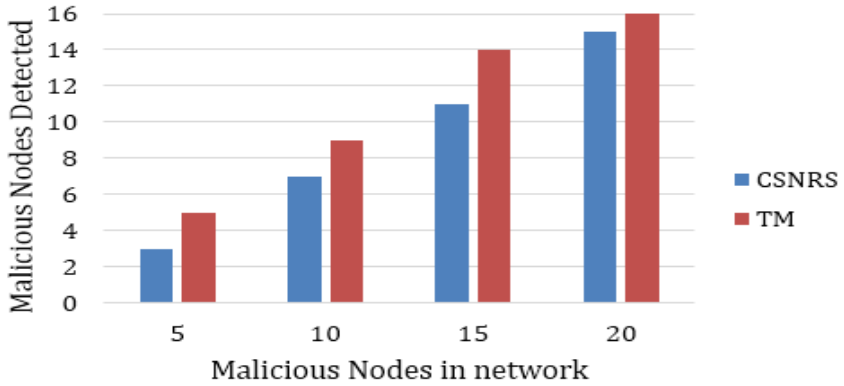

Figure 8. Comparative analysis for second case study (University Car Parking)

Table 8. Parameters for the third case study (Traffic Monitoring) experimentation

Parameters	Values
End Devices	15
Gate Way	1
Number of Malicious Nodes	1 - 7

Table 9. TM Confusion Matrix for Case Study 3 (Traffic Monitoring)

TM Confusion Matrix		Predicted		
Actual		Non-Malicious	Malicious	
	Non-Malicious	[True Negative] 44	[False Positive] 0	44
	Malicious	[False Negative] 3	[True Positive] 13	16
		47	13	60

Table 10. CSNRS Confusion Matrix for Case Study 3 (Traffic Monitoring)

CSNRS Confusion Matrix		Predicted		
Actual		Non-Malicious	Malicious	
	Non-Malicious	[True Negative] 44	[False Positive] 0	44
	Malicious	[False Negative] 10	[True Positive] 6	16
		54	6	60

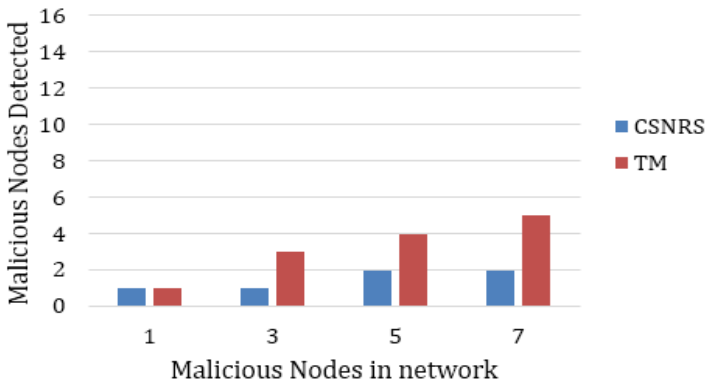

Figure 9. Comparative analysis for the third case study (Traffic Monitoring)

Discussion and Analysis of Results:

The above tests showed the number of malicious nodes detected correctly by both techniques. The results of this case study show that the network size varies from fifteen to three fifty and we have one to twenty malicious nodes in our experiments. The proposed TM methodology detected more malicious nodes as compared to the CSNRS technique because we only assigned trust values to an end device based on the interactions observed by its neighbor end devices. While in the counteracting selfish nodes using CSNRS, by default every end device was considered a good node by assigning the trust value of 0.5. This 0.5 value is the threshold value and the end device is not malicious when the trust value is greater than 0.5, even when the packets are not exchanged between the end device and gateway. Therefore, when the overall trust value of the end device is calculated by the CSNRS technique, some malicious end devices are considered as not malicious because the rest of the end devices in the network, who were not neighbors of those malicious end devices, didn't vote against it. It is also observed that some end devices only have one or two neighbors, so their trust rating did not significantly affect the overall trust value of a malicious end device. Furthermore, the Subjective Logic metrics delivered better results without consuming excessive computing resources on an end device.

According to the results, our proposed technique has more ability to detect malicious nodes as compared to the CSNRS technique. Moreover, when we increase our network size, the ability to detect the malicious node of the TM technique is greater as compared to the CSNRS technique. When the network is denser, both of the techniques were able to detect the majority of the malicious end devices. But if the number of end devices is less in number, then the ability of the CSNRS was reduced to detect the malicious end devices. Therefore, it can be observed that TM has more accuracy and less error rate as compared to CSNRS.

Conclusion and Recommendations:

The article presents a trust management technique to detect malicious attacks such as denial of services attacks, black hole attacks, and selective forwarding attacks in our LPWAN network for Smart City applications. It can be observed that our proposed TM technique that is presented in this work can detect malicious nodes better than the CSNRS technique. LPWAN can send data at a long distance and can assure more battery timings. Our assumptions for LPWAN are sufficient for general deductions. However, this method can be highly useful for multiple applications and can compete with other data transfer techniques. Trust management can also be used for other data transfer protocols, unlike LPWAN which can be examined and implemented later on. In future work, we can also deploy the solution using LPWAN devices and test the authenticity of the simulation results.

References:

[1] M. Angelidou, "Smart city policies: A spatial approach," Cities, vol. 41, pp. S3–S11, Jul. 2014, doi: 10.1016/J.CITIES.2014.06.007.

[2] P. Neirotti, A. De Marco, A. C. Cagliano, G. Mangano, and F. Scorrano, "Current trends in Smart City initiatives: Some stylised facts," Cities, vol. 38, pp. 25–36, Jun. 2014, doi: 10.1016/J.CITIES.2013.12.010.

[3] D. Belanche, L. V. Casaló, and C. Orús, "City attachment and use of urban services: Benefits for smart cities," Cities, vol. 50, pp. 75–81, Feb. 2016, doi: 10.1016/J.CITIES.2015.08.016.

[4] A. Caragliu, C. del Bo, and P. Nijkamp, "Smart cities in Europe," J. Urban Technol., vol. 18, no. 2, pp. 65–82, Apr. 2011, doi: 10.1080/10630732.2011.601117.

[5] M. P. Robertas Jucevičius, Irena Patašienė, "Digital Dimension of Smart City: Critical Analysis," Procedia - Soc. Behav. Sci., vol. 156, no. 26, pp. 146–150, 2014, doi: https://doi.org/10.1016/j.sbspro.2014.11.137.

[6] P.P. Ray, "A survey on Internet of Things architectures," J. King Saud Univ. - Comput. Inf. Sci., vol. 30, no. 3, pp. 291–319, 2018, doi: https://doi.org/10.1016/j.jksuci.2016.10.003.

[7] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low Power Wide Area Networks: An Overview," IEEE Commun. Surv. Tutorials, vol. 19, no. 2, pp. 855–873, Apr. 2017, doi: 10.1109/COMST.2017.2652320.

[8] N. K. S. Aggarwal, "Attacks on Blockchain Working model," Adv. Comput. Elsevier, vol. 121, 2021.

[9] J. Bennaceur, H. Idoudi, and L. Azouz Saidane, "Trust management in cognitive radio networks: A survey," Int. J. Netw. Manag., vol. 28, no. 1, Jan. 2018, doi: 10.1002/NEM.1999.

[10] "Leader in Cybersecurity Protection & Software for the Modern Enterprises - Palo Alto Networks." Accessed: Feb. 25, 2025. [Online]. Available: https://www.paloaltonetworks.com/

[11] "(PDF) A Study on Black Hole Attack in Wireless Sensor Networks." Accessed: Feb. 25, 2025. [Online]. Available: https://www.researchgate.net/publication/317339617_A_Study_on_Black_Hole_Attack_in_Wireless_Sensor_Networks

[12] "(PDF) Routing Attacks in Wireless Sensor Networks: A Survey." Accessed: Feb. 25, 2025. [Online]. Available: https://www.researchgate.net/publication/263967791_Routing_Attacks_in_Wireless_Sensor_Networks_A_Survey

[13] Z. Banković, J. C. Vallejo, D. Fraga, and J. M. Moya, "Detecting Bad-Mouthing Attacks on Reputation Systems Using Self-Organizing Maps," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 6694 LNCS, pp. 9–16, 2011, doi: 10.1007/978-3-642-21323-6_2.

[14] P. N. C. AMOL R. DHAKNE, "DETECTION OF ON-OFF ATTACK BASED ON PREDECTABILITY TRUST IN WIRELESS SENSOR NETWORK," Int. J. Adv. Comput. Eng. Netw., vol. 4, no. 12, 2016, [Online]. Available: https://www.iraj.in/journal/journal_file/journal_pdf/3-325-148463785729-33.pdf

[15] "Ekran - Wikipedia." Accessed: Feb. 25, 2025. [Online]. Available: https://en.wikipedia.org/wiki/Ekran

[16] L. K. Bysani and A. K. Turuk, "A survey on selective forwarding attack in wireless sensor networks," 2011 Int. Conf. Devices Commun. ICDeCom 2011 - Proc., 2011, doi: 10.1109/ICDECOM.2011.5738547.

[17] "(PDF) A Survey on Detection of Sinkhole Attack in Wireless Sensor Network." Accessed: Feb. 25, 2025. [Online]. Available: https://www.researchgate.net/publication/282024860_A_Survey_on_Detection_of_Sinkhole_Attack_in_Wireless_Sensor_Network

[18] S. Qazi, R. Raad, Y. Mu, and W. Susilo, "Securing DSR against wormhole attacks in multirate ad hoc networks," J. Netw. Comput. Appl., vol. 36, no. 2, pp. 582–592, Mar. 2013, doi: 10.1016/J.JNCA.2012.12.019.

[19] Rupali Sharma, "Gray-hole Attack in Mobile Ad-hoc Networks : A Survey," Rupali Sharma / Int. J. Comput. Sci. Inf. Technol., vol. 7, no. 3, pp. 1457–1460, 2016, [Online]. Available: https://www.ijcsit.com/docs/Volume 7/vol7issue3/ijcsit2016070389.pdf

[20] "Investopedia." Accessed: Feb. 25, 2025. [Online]. Available: https://www.investopedia.com/

[21] "OWASP - Wikipedia." Accessed: Feb. 25, 2025. [Online]. Available: https://en.wikipedia.org/wiki/OWASP

[22] Z. A. Khan and P. Herrmann, "A trust based distributed intrusion detection mechanism for internet of things," Proc. - Int. Conf. Adv. Inf. Netw. Appl. AINA, pp. 1169–1176, May 2017, doi: 10.1109/AINA.2017.161.

[23] J. Gao, W. Xu, S. Kanhere, S. Jha, and W. Hu, "Poster abstract: A novel modeling involved security approach for lora key generation," Proc. - 2020 19th ACM/IEEE Int. Conf. Inf. Process. Sens. Networks, IPSN 2020, pp. 327–328, Apr. 2020, doi: 10.1109/IPSN48710.2020.00-23.

[24] S. L. K. I. Ahmad, K. -L. A. Yau, M. H. Ling, "Trust and Reputation Management for Securing Collaboration in 5G Access Networks: The Road Ahead," IEEE Access, vol. 8, pp. 62542–62560, 2020, doi: 10.1109/ACCESS.2020.2984318.

[25] V. Ribeiro, R. H. Filho, and A. Ramos, "A Secure and Fault-Tolerant Architecture for LoRaWAN Based on Blockchain," 2019 3rd Cyber Secur. Netw. Conf. CSNet 2019, pp. 35–41, Oct. 2019, doi: 10.1109/CSNET47905.2019.9108933.

[26] S. U. J. K. A. Awan, I. U. Din, M. Zareei, M. Talha, M. Guizani, "HoliTrust-A Holistic Cross-Domain Trust Management Mechanism for Service-Centric Internet of Things," IEEE Access, vol. 7, pp. 52191–52201, 2019, doi: 10.1109/ACCESS.2019.2912469.

[27] S. U. J. K. A. Awan, I. Ud Din, A. Almogren, M. Guizani, A. Altameem, "RobustTrust – A Pro-Privacy Robust Distributed Trust Management Mechanism for Internet of Things," IEEE Access, vol. 7, pp. 62095–62106, 2019, doi:

10.1109/ACCESS.2019.2916340.

[28] K. Harsanyi, A. Kiss, and T. Sziranyi, "Wormhole detection in wireless sensor networks using spanning trees," 2018 IEEE Int. Conf. Futur. IoT Technol. Futur. IoT 2018, vol. 2018-January, pp. 1–6, Mar. 2018, doi: 10.1109/FIOT.2018.8325596.

[29] X. S. ye Zhengwang, Tao Wen, Zhenyu Liu, "An Efficient Dynamic Trust Evaluation Model for Wireless Sensor Networks," J. Sensors, vol. 2, pp. 1–16, 2017, doi: 10.1155/2017/7864671.

[30] N. Alsaedi, F. Hashim, A. Sali, and F. Z. Rokhani, "Detecting sybil attacks in clustered wireless sensor networks based on energy trust system (ETS)," Comput. Commun., vol. 110, pp. 75–82, Sep. 2017, doi: 10.1016/J.COMCOM.2017.05.006.

[31] J. H. P. D. -Y. Kim, S. Kim, "Remote Software Update in Trusted Connection of Long Range IoT Networking Integrated With Mobile Edge Cloud," IEEE Access, vol. 6, pp. 66831–66840, 2018, doi: 10.1109/ACCESS.2017.2774239.

[32] M. K. H. Hellaoui, A. Bouabdallah, "TAS-IoT: Trust-Based Adaptive Security in the IoT," 2016 IEEE 41st Conf. Local Comput. Networks (LCN), Dubai, United Arab Emirates, pp. 599–602, 2016, doi: 10.1109/LCN.2016.101.

[33] E. K. Wang, T. Y. Wu, C. M. Chen, Y. Ye, Z. Zhang, and F. Zou, "MDPAS: Markov Decision Process based adaptive security for sensors in Internet of things," Adv. Intell. Syst. Comput., vol. 329, pp. 389–397, 2015, doi: 10.1007/978-3-319-12286-1_40.

[34] I. R. Chen, J. Guo, and F. Bao, "Trust Management for SOA-Based IoT and Its Application to Service Composition," IEEE Trans. Serv. Comput., vol. 9, no. 3, pp. 482–495, May 2016, doi: 10.1109/TSC.2014.2365797.

[35] S. A. Muhammad Fayaz, Ajab Khan, Gulzar Mehmood, "Counteracting Selfish Nodes Using Reputation Based System in Mobile Ad Hoc Networks," Electronics, 2022, doi: 10.3390/electronics11020185.

[36] P. Herrmann, "Temporal logic-based specification and verification of trust models," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 3986 LNCS, pp. 105–119, 2006, doi: 10.1007/11755593_9.