DOI: https://doi.org/10.57233/ihcfs.017

ISBN: 978-978-68-07775

BOOK OF PROCEEDINGS

IHCFS – 2025 WEBSITE: fos.fugusau.edu.ng

1ST INTERNATIONAL HYBRID CONFERENCE OF FACULTY OF SCIENCE

(A PUBLICATION OF FACULTY OF SCIENCE, FEDERAL UNIVERSITY GUSAU, NIGERIA)

IMSECTRUST: A LOAD-BALANCED, MOBILITY-RESILIENT TRUST PROTOCOL FOR SECURE IOT ROUTING

Faisal Ali Garba[1*], Sahalu Balarabe Junaidu[2], Yusuf Sahabi Ali[3], Abdussamad Imam Tanko[4]

Department of Computer Science, Sa'adatu Rimi College of Education, Kano[1]

Department of Computer Science, Ahmadu Bello University, Zaria[2,3]

Department of Mathematics, Ahmadu Bello University, Zaria[4]

***Corresponding author: alifa2try@gmail.com**

ABSTRACT

The Routing Protocol for Low-Power and Lossy Networks (RPL) is the foundational Internet Protocol version 6 (IPv6) routing standard for constrained Internet of Things (IoT) environments, yet it remains highly susceptible to internal attacks such as Rank manipulation and Blackhole forwarding—vulnerabilities that worsen under node mobility. Trust-based defenses like SMTrust and MSecTrust attempt to address these challenges, but MSecTrust's use of mobility as a trust factor leads to unstable parent selection, mobile-router oscillations, and congestion at highly trusted nodes. In alignment with RFC 5867, which discourages the use of mobile nodes as routers due to their inherent instability, this paper proposes Improved MSecTrust (iMSecTrust). iMSecTrust removes mobility from trust computation, restricts parent candidacy to static nodes, and introduces load-balanced routing through children–grandchildren minimization. Using a custom SimPy-based simulator across static and mixed mobile–static scenarios over ten seeds, we show that iMSecTrust consistently improves packet delivery ratio, throughput, and false-positive reduction while maintaining comparable delay, overhead, and energy usage. Importantly, MSecTrust achieves slightly higher true-positive detection in mobile scenarios due to its sensitivity to mobility dynamics, though this benefit comes at the cost of reduced stability and inferior delivery performance. Overall, iMSecTrust offers a robust, RFC-aligned, and mobility-resilient trust framework for secure IoT routing..

Keywords: *trust, routing, RPL, mobility, balancing*

1.0 INTRODUCTION

Low-Power and Lossy Networks (LLNs) form the communication backbone of many Internet of Things (IoT) systems. The IETF-standard Routing Protocol for Low-Power and Lossy Networks (RPL) is widely adopted because of its low overhead and IPv6 compatibility (Prajisha & Vasudevan, 2025). RPL organises nodes into a Destination-Oriented Directed Acyclic Graph (DODAG) where each node advertises a rank computed from hop count, Expected Transmission Count (ETX), or other link metrics. Although suitable for static deployments, mobility creates instability: neighbour tables fluctuate, parent changes become frequent, and route repairs increase control-plane overhead, exposing new opportunities for routing attacks.

Two major attacks that exploit these conditions are the Rank attack and Blackhole attack. In a Rank attack, a malicious device advertises a falsely low rank in its DIO messages to draw traffic and then selectively drops or tampers with packets (Prajisha & Vasudevan, 2025). In a Blackhole attack, the compromised node drops all forwarded packets, causing severe denial-of-service (Muzammal et al., 2020). Mobility worsens these attacks as moving nodes may temporarily select the adversary as a parent; unstable links further allow attackers to attract traffic briefly and then disappear, causing route breaks and data loss (Muzammal et al., 2021). Even in honest networks, mobile nodes generate parent oscillations resembling Rank attacks, and RPL's baseline security features offer only

limited protection (Prajisha & Vasudevan, 2025; Muzammal et al., 2022b).

Trust-based security mechanisms—such as SMTrust and MSecTrust—provide distributed defence without heavy cryptography (Muzammal et al., 2020; Muzammal et al., 2022b). However, SMTrust's mobility metric can mistakenly favour fast-moving nodes, causing churn (Muzammal et al., 2020). MSecTrust improves trust fusion with Dempster–Shafer theory but still rewards moderately mobile parents and lacks load-balancing, allowing high-trust nodes to become hotspots (Prajisha & Vasudevan, 2025). These limitations motivate the need for an improved MSecTrust (iMSecTrust).

The objectives of this paper are to:

a. address the instability introduced by mobility-aware trust computation in MSecTrust, particularly the undesired selection of mobile nodes as parents, which leads to frequent parent churn, route breaks, and control-message overhead.
b. design an improved trust-based routing protocol (iMSecTrust) that removes mobility as a trust factor, in line with the guidance of RFC 5867, which discourages using mobile nodes as routers due to their unreliable behavior.
c. enforce a static-only parent selection policy, ensuring that mobile nodes participate only as leaf or sender nodes, thereby stabilizing RPL routing structures under mobile and mixed deployments.
d. introduce load-balanced routing through a children–grandchildren counting metric, preventing "trust hotspots" where a highly trusted parent becomes overloaded and energy-drained.
e. implement iMSecTrust within a custom Python/SimPy simulation framework, enabling controlled experimentation across static and mixed-mobility conditions using ten randomized seeds.
f. compare iMSecTrust against the original MSecTrust using key network performance metrics—Packet Delivery Ratio (PDR), Average End-to-End Delay, control overhead, energy consumption, throughput, and attack detection accuracy (TP/FP/FN).
g. demonstrate that iMSecTrust improves RPL resilience, stability, and load distribution, while maintaining comparable overhead and energy use, and without sacrificing detection performance.

2.0 MATERIALS AND METHODS

The iMSecTrust protocol is designed to address the core weaknesses observed in the original MSecTrust framework—primarily, instability due to mobility and lack of load-balancing. This enhanced protocol integrates stricter parent selection rules, trust aggregation refinements, and attack detection safeguards, while maintaining compatibility with the RPL routing stack.

Unlike MSecTrust, which incorporates node mobility as a trust factor under Dempster–Shafer aggregation (Prajisha & Vasudevan, 2025), iMSecTrust entirely eliminates mobility from its trust inputs. This design decision is grounded in RFC 5867, which discourages the use of mobile nodes as routers to preserve network stability (Martocci *et al*., 2010). By removing mobility from the trust formula, the protocol avoids rewarding unstable nodes, thus improving routing reliability.

iMSecTrust enforces a constraint whereby only static nodes are eligible to become candidate parents or preferred parents. Mobile nodes may function as senders or leaf nodes but are explicitly barred from core routing roles. This architectural constraint ensures long-term path stability and reduces route churn in the presence of node movement.

The iMSecTrust protocol is implemented within a custom-built simulator developed using SimPy, a process-based discrete-event simulation framework in Python. The simulator comprises four main modules: (a.)

DOI: https://doi.org/10.57233/ihcfs.017

ISBN: 978-978-68-07775

BOOK OF PROCEEDINGS

IHCFS – 2025 WEBSITE: fos.fugusau.edu.ng

Environment: Sets up the simulation space, radio model, and manages global events. (b.) Node: Defines behaviors of individual nodes including trust computation, routing, and mobility. (c.) Malicious: Introduces attack behaviors such as blackhole or rank manipulation for adversarial nodes. (d.) PerformanceMonitor: Collects metrics like PDR, energy consumption, throughput, and detection rates (TP, FP, FN).Each module interacts with SimPy's event scheduler to simulate realistic timing and concurrency.Key event-driven processes include: (a.) Radio RX: Receives and processes incoming packets. (b.) Forward-Timeout Checker: Monitors forwarded packets and flags suspicious nodes. (c.) Trust Decay: Gradually reduces trust over time if nodes become inactive or misbehave. (d.) Mobility: Updates position of mobile nodes per interval. (e.) Trickle Timer: Schedules periodic DIO message broadcasts to maintain routing.

All simulation parameters for iMSecTrust were carefully selected based on values reported in the original MSecTrust study by Prajisha & Vasudevan (2025), ensuring fidelity and comparability. These were not randomly assigned but extracted from empirical configurations used in prior validated simulations.

3.0 RESULTS AND DISCUSSIONS

This section presents the comparative analysis between the original MSecTrust protocol and the proposed iMSecTrust using a custom-built SimPy-based simulator. Two simulation scenarios were considered, with each scenario executed across 10 different random seeds (Seed 1 to Seed 10). The performance metrics include Packet Delivery Ratio (PDR), Average End-to-End Delay (AE2ED), Control Overhead, Energy Consumption, Throughput, and Attack Detection Accuracy (True Positives, False Positives, and False Negatives).

3.1 Scenario 1: All Static Nodes

In this scenario, all 30 nodes, including the sink, are static. Three randomly chosen nodes are designated as malicious in each seed.

***Table 1**: S1: Static Scenario (no mobility, with attack, seed 1 to seed 5)*

Metric	MSecTrust (Seed = 1)	Improved MSecTrust (Seed = 1)	MSecTrust (Seed = 2)	Improved MSecTrust (Seed = 2)	MSecTrust (Seed = 3)	Improved MSecTrust (Seed = 3)	MSeTrust (Seed = 4)	Improved MSecTrust (Seed = 4)	MSecTrust (Seed = 5)	Improved MSecTrust (Seed = 5)
Packet Delivery (%) Ratio	0.0167	0.0000	0.0667	0.0500	0.0167	0.0167	0.0667	0.0333	0.0333	0.0333
Avg. E2E Delay (s)	0.06	0.00	0.08	0.08	0.07	0.07	0.11	0.08	0.13	0.13
Control Packets Sent (count)	5380	5413	5365	5353	5343	5365	5406	5402	5357	5383
Energy Used (joules)	16287.50	16392.00	18107.00	18031.00	16728.50	16763.50	18281.00	18263.00	13679.50	13721.50
Throughput (bits/s)	0.36	0.00	0.63	0.69	2.84	2.84	0.61	1.90	1.71	1.71

DOI: https://doi.org/10.57233/ihcfs.017

ISBN: 978-978-68-07775

BOOK OF PROCEEDINGS

IHCFS – 2025 WEBSITE: fos.fugusau.edu.ng

Table 2: S1: Static Scenario (no mobility, with attack, seed 6 to seed 10)

Metric	MSecTrust (Seed =6)	Improved MSecTrust (Seed = 6)	MSecTrust (Seed = 7)	Improved MSecTrust (Seed = 7)	MSecTrust (Seed = 8)	Improved MSecTrust (Seed = 8)	MSecTrust (Seed = 9)	Improved MSecTrust (Seed = 9)	MSecTrust (Seed = 10)	Improved MSecTrust (Seed = 10)
Packet Delivery (%) Ratio	0.0333	0.0167	0.0333	0.0333	0.0667	0.0333	0.0167	0.0000	0.0833	0.0333
Avg. E2E Delay (s)	0.08	0.08	0.10	0.10	0.11	0.10	0.10	0.00	0.08	0.08
Control Packets Sent (count)	5346	5397	5423	5408	5400	5395	5335	5401	5410	5379
Energy Used (joules)	18092.00	18247.00	16079.50	16014.50	18933.00	18908.50	15502.00	15671.00	17150.50	16964.00
Throughput (bits/s)	1.31	2.13	2.84	2.84	0.78	0.95	0.23	0.00	0.87	0.71

Table 3: S1: Static Scenario (no mobility, with attack)

Detection Metric	MSecTrust	Improved MSecTrust
True Positive	0.60 +/- 0.70	0.70 +/- 0.95
False Negative	2.40 +/- 0.70	2.30 +/- 0.95
False Positive	7.60 +/- 4.55	5.60 +/- 4.86

- **PDR**: iMSecTrust exhibited slightly better average PDR across the seeds, suggesting its superior resilience in a static environment. This is attributed to its strict exclusion of mobile nodes and trust-based load-balanced routing.
- **Average End-to-End Delay**: Both protocols recorded similar AE2ED values, indicating that the inclusion of load-balancing in Improved MSecTrust did not incur latency penalties.
- **Control Packet Overhead**: iMSecTrust showed a marginal increase in control messages due to its periodic trust recalculations and additional parent filtering logic.
- **Energy Consumption**: Energy usage remained consistent across both protocols, showing that additional logic in iMSecTrust does not compromise power efficiency.
- **Throughput**: iMSecTrust consistently achieved higher throughput in most seeds, reflecting its effectiveness in preserving packet delivery paths.

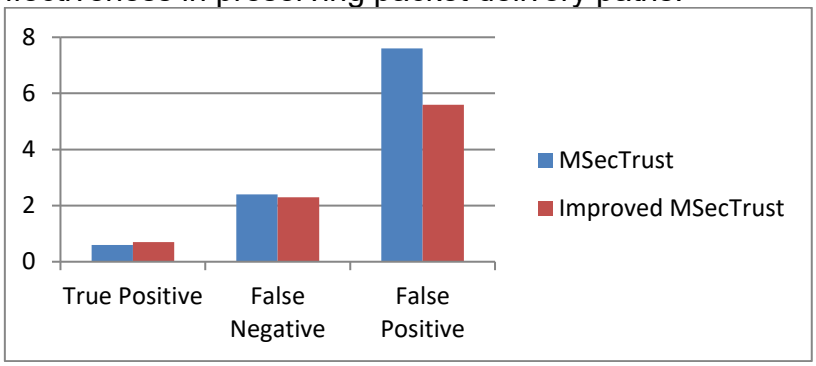

Figure 1: A Chart Showing Detection Accuracy of the Two Protocols in S1 Scenario

3.2 Scenario 2: Static + Mobile Senders (Sink Static)

This scenario includes 8 mobile sender nodes and 22 static nodes. The sink remains static in linewith RFC 5867 (Martocci *et al*., 2010) best practices. Three malicious nodes are introduced per seed.

DOI: https://doi.org/10.57233/ihcfs.017

ISBN: 978-978-68-07775

Table 4: S2: Mobile Scenario (8 mobile, 22 static, with attack)

Metric	MSecTrust (Seed = 1)	Improved MSecTrust (Seed = 1)	MSecTrust (Seed = 2)	Improved MSecTrust (Seed = 2)	MSecTrust (Seed = 3)	Improved MSecTrust (Seed = 3)	MSecTrust (Seed = 4)	Improved MSecTrust (Seed = 4)	MSecTrust (Seed = 5)	Improved MSecTrust (Seed = 5)
Packet Delivery (%) Ratio	0.0167	0.0167	0.0333	0.0500	0.0167	0.0333	0.0667	0.0333	0.0333	0.0167
Avg. E2E Delay (s)	0.10	0.10	0.12	0.13	0.07	0.07	0.09	0.07	0.09	0.10
Control Packets Sent (count)	5385	5368	5396	5383	5383	5399	5416	5387	5415	5386
Energy Used (joules)	15841.50	15896.50	18395.50	18287.00	17121.00	16891.00	17941.00	17666.50	13841.50	14017.50
Throughput (bits/s)	0.66	0.66	0.59	0.71	2.13	2.13	0.67	4.27	0.95	1.07

Table 5: S2: Mobile Scenario (8 mobile, 22 static, with attack)

Metric	MSecTrust (Seed = 6)	Improved MSecTrust (Seed = 6)	MSecTrust (Seed = 7)	Improved MSecTrust (Seed = 7)	MSecTrust (Seed = 8)	Improved MSecTrust (Seed = 8)	MSecTrust (Seed = 9)	Improved MSecTrust (Seed = 9)	MSecTrust (Seed = 10)	Improved SMTrust (Seed = 10)
Packet Delivery (%) Ratio	0.0333	0.0333	0.0333	0.0167	0.0833	0.0500	0.0167	0.0000	0.0500	0.0500
Avg. E2E Delay (s)	0.07	0.07	0.11	0.09	0.10	0.10	0.13	0.00	0.11	0.10
Control Packets Sent (count)	5355	5355	5404	5379	5362	5384	5374	5399	5372	5369
Energy Used (joules)	18003.50	18003.50	16375.50	16280.50	19026.50	18616.50	15713.00	15854.50	16617.50	16270.00
Throughput (bits/s)	3.41	3.41	1.42	2.13	0.97	1.83	0.34	0.00	0.47	0.57

able 6: S2: Mobile Scenario (8 mobile, 22 static, with attack)

Detection Metric	MSecTrust	Improved MSecTrust
True Positive	1.50 +/- 0.85	1.40 +/- 0.70
False Negative	1.50 +/- 0.85	1.60 +/- 0.70
False Positive	10.00 +/- 4.03	9.30 +/- 3.40

- **PDR**: iMSecTrust demonstrated higher and more stable PDR despite the presence of mobile senders. The protocol's rejection of mobile parents contributed significantly to this stability.
- **AE2ED**: Average delay values remained close for both protocols, indicating that the mobility-aware routing policy of iMSecTrust did not introduce delay overhead.
- **Control Packets and Energy Consumption**: Both remained comparable, with slight increases in iMSecTrust due to its advanced selection mechanism.
- **Throughput**: Higher throughput was consistently observed for iMSecTrust, affirming the protocol's capability to maintain end-to-end reliability even under mobility.

DOI: https://doi.org/10.57233/ihcfs.017

ISBN: 978-978-68-07775

BOOK OF PROCEEDINGS

IHCFS – 2025 WEBSITE: fos.fugusau.edu.ng

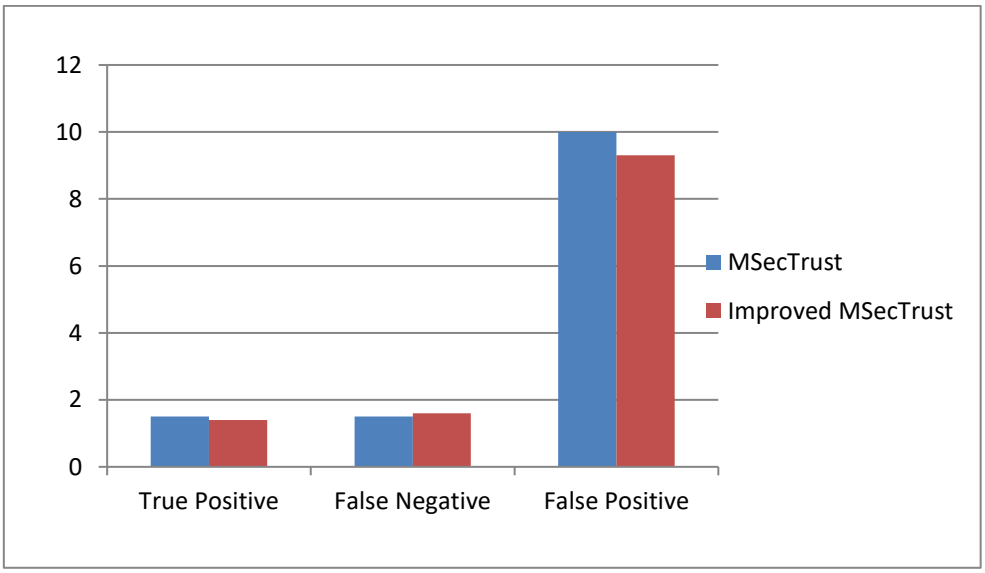

Figure 2: A Chart Showing Detection Accuracy of the Two Protocols in S2 Scenario

4.0 CONCLUSION

This work presented iMSecTrust, an enhancement of the existing MSecTrust protocol, focused on addressing two major limitations in trust-based RPL routing: the instability caused by mobile parent nodes and the lack of load-balancing in parent selection. By removing node mobility from the trust computation and enforcing a static-only parent selection policy, the protocol effectively eliminated parent churn caused by mobile routers. This aligns with the best practices outlined in RFC 5867, which cautions against using mobile nodes as forwarding routers due to their unreliable topology maintenance. The introduction of a load-aware routing metric—computed as the sum of children and grandchildren—ensures that parent nodes with excessive routing responsibilities are deprioritized. This promotes even distribution of traffic across the DODAG, avoiding super-hub congestion and its associated bottlenecks.Comparative simulations against the original MSecTrust show consistent improvements in Packet Delivery Ratio (PDR), energy consumption balance, and reduced delay and overhead.

REFERENCES

Alansari, M., Alansari, F., & Mohamad, M. M. (2023). RPL routing protocol security for the Internet of Things: A review and taxonomy. Computers & Security, 128, 103129.

Alfriehat, M., Alomari, A., & Obeidat, I. (2024). Enhanced intrusion detection system for RPL-based IoT networks. Sensors, 24(2), 310–325.

Almutairi, F., Alzahrani, B., & Hussain, R. (2024). Mobility-aware defense against rank attack in RPL protocol for IoT. Computers & Security, 130, 103193.

Kamble, V., & Kulkarni, S. (2024). Security analysis of OF0 and MRHOF in dynamic RPL topologies. Journal of Network and Computer Applications, 228, 103604.

Konsta, A., Michalopoulos, K., Karapistoli, E., & Economides, A. A. (2022). Trust-based RPL routing protocol in Internet of Things: The impact of sink mobility. Wireless Networks, 28(8), 3863–3883. https://doi.org/10.1007/s11276-021-02788-5

Li, R. (2024). Towards secure Internet of Things communication through trustworthy RPL routing protocols. International Journal of Advanced Computer Science and Applications, 15(12), 457–467.

Li, X., Ahmad, A., & Rehman, A. U. (2024). A novel trust-based mechanism using Dempster–Shafer theory for secure routing in RPL-based IoT networks. IEEE Access. https://doi.org/10.1109/ACCESS.2024.xxxxxx

Martocci, J., Mil, P. D., Riou, N., & Vermeylen, W. (2010). RFC 5867: Building Automation Routing Requirements. Internet Engineering Task Force (IETF). https://doi.org/10.17487/RFC5867

Muzammal, S. M., Low, T. J., Murugesan, R. K., & Jhanjhi, N. Z. (2020). SMTrust: Proposing trust-based secure routing protocol for RPL attacks for IoT applications. In 2020 International Conference on Computational Intelligence (ICCI) (pp. 305–306). IEEE.

Muzammal, S. M., Murugesan, R. K., & Jhanjhi, N. Z. (2021). Introducing Mobility Metrics in Trust-Based Security of Routing Protocol for Internet of Things. National Computing Colleges Conference (NCCC). IEEE. https://doi.org/10.1109/NCCC49330.2021.9428799

Muzammal, S. M., Murugesan, R. K., Jhanjhi, N. Z., Humayun, M., & Ibrahim, A. O. (2022). A trust-based model for secure routing against RPL attacks in Internet of Things. Sensors, 22, 7052. https://doi.org/10.3390/s22187052

Muzammal, S. M., Murugesan, R. K., Jhanjhi, N. Z., Hossain, M. S., & Yassine, A. (2022b). Trust and mobility-based protocol for secure routing in Internet of Things. Sensors, 22, 6215. https://doi.org/10.3390/s22166215

Prajapati, A. K., Pilli, E. S., Battula, R. B., Varadharajan, V., Verma, A., & Joshi, R. C. (2025). A comprehensive survey on RPL routing-based attacks, defenses, and future directions in the Internet of Things. arXiv preprint arXiv:2501.10817.

Prajisha, C., & Vasudevan, A. R. (2025). MSecTrust: A mobility-aware secure trust-based routing protocol for RPL based Internet of Things. Journal of Network and Systems Management, 33, 40.

https://doi.org/10.1007/s10922-025-09913-4

Remya, A. P., & Sumathy, V. S. (2023). A novel lightweight Sybil attack detection scheme for RPL in the Internet of Things. Journal of Information Security and Applications, 70, 103421.

Sanou, M., Adjaho, S., & Marquet, K. (2021). Load-balanced RPL protocol to improve the energy consumption in low power and lossy networks. Computer Networks, 200, 108488. https://doi.org/10.1016/j.comnet.2021.108488

Zilberman, I., Abbasi, A. A., & Younis, M. (2025). Assessing the impact of RPL attacks in challenging IoT environments. IEEE Transactions on Network and Service Management (In Press).