Trust-Based Security Architecture for Edge Computing: A Simulation Study of Dynamic Trust Evolution and Attack Detection

Oleksandr Kuznetsov[1,*,†], Emanuele Frontoni[2,†], Yelyzaveta Kuznetsova[3,†], Oleksii Smirnov[4,†] and Illarion Moskovchenko[5,†]

[1] *Department of Theoretical and Applied Sciences, eCampus University, Via Isimbardi 10, Novedrate (CO), 22060, Italy*
[2] *Department of Political Sciences, Communication and International Relations, University of Macerata, Via Crescimbeni, 30/32, 62100 Macerata, Italy*
[3] *Department of Information and Communication Systems Security, School of Computer Sciences, V. N. Karazin Kharkiv National University, 4 Svobody Sq., 61022 Kharkiv, Ukraine*
[4] *Department of cyber security and software, Central Ukrainian National Technical University, 8, University Ave, Kropyvnytskyi, 25006, Ukraine*
[5] *Department of mathematical and software of automated control systems, Faculty of automated control systems and ground support for aviation flights, Ivan Kozhedub Kharkiv National Air Force University, Sumska str., 77/79, Kharkiv, 61023, Ukraine*

Abstract

This paper presents a comprehensive experimental study of a novel trust-based security architecture for edge computing environments. We introduce an adaptive security framework that combines dynamic trust evaluation with decentralized decision-making mechanisms to enhance threat detection and system resilience. Through extensive simulation experiments, we evaluate the architecture's performance across various network configurations, ranging from 20 to 100 nodes, with different operational parameters and security event patterns. The simulation framework implements a sophisticated spatial distribution model for edge nodes, incorporating computational constraints, memory limitations, and communication boundaries typical of edge computing environments. Our results demonstrate that the proposed architecture achieves an 83.0% threat detection rate while maintaining network resilience at 95.6%, significantly exceeding baseline security requirements. The trust management mechanism demonstrates robust adaptation to security events, maintaining average trust scores of 78.6% despite active security incidents. We provide detailed analysis of system behavior under various attack scenarios, including intrusion attempts, data leaks, DDoS attacks, and authentication failures. The architecture shows exceptional scalability characteristics, with improved detection rates and trust stability in larger network configurations. Performance metrics reveal consistent achievement above target thresholds across all evaluated dimensions, with minimum trust levels maintaining a 7.2 percentage point margin above requirements. Our findings provide empirical validation of the architecture's effectiveness while offering practical insights into deployment considerations for edge computing security. The study contributes to the field by establishing quantitative benchmarks for security performance in edge environments and demonstrating the viability of trust-based security mechanisms for distributed systems.

Keywords

edge computing security, trust evolution simulation, security architecture evaluation, dynamic trust management, attack detection mechanisms, network resilience analysis, security performance metrics, distributed security systems, adaptive security framework, edge computing simulation

Information Technology and Implementation (IT&I-2024), November 20-21, 2024, Kyiv, Ukraine
* Corresponding author.
† These authors contributed equally.
oleksandr.kuznetsov@uniecampus.it (O. Kuznetsov); emanuele.frontoni@unimc.it (E. Frontoni); elizabet8smidt12@gmail.com (Y. Kuznetsova); dr.SmirnovOA@gmail.com (O. Smirnov); illarion_moskovchenko@ukr.net (I. Moskovchenko)
0000-0003-2331-6326 (O. Kuznetsov); 0000-0002-8893-9244 (E. Frontoni); 0000-0002-0573-0913 (Y. Kuznetsova); 0000-0001-9543-874X (O. Smirnov); 0000-0002-7058-0691 (I. Moskovchenko)

1. Introduction

Edge computing represents a fundamental transformation in distributed system architectures, shifting computational resources closer to data sources and end devices [1,2]. This paradigm has emerged as a critical enabler for latency-sensitive applications and real-time data processing, particularly in domains such as Industrial IoT, smart cities, and healthcare systems [3,4]. However, the distributed nature of edge computing introduces complex security challenges that traditional centralized security approaches fail to address adequately [5].

Recent industry analyses project that edge devices will generate over 79.4 ZB of data by 2025, with approximately 75% of enterprise data being processed at the edge [6]. This massive decentralization of computation creates unprecedented security vulnerabilities. Edge nodes often operate in untrusted environments, face resource constraints, and must handle dynamic network conditions while maintaining robust security guarantees. These challenges are compounded by the heterogeneous nature of edge devices and their diverse operational requirements.

Traditional security architectures, designed for centralized cloud environments, prove inadequate in edge computing scenarios for several reasons [7]. First, they typically assume stable network connectivity and abundant computational resources, assumptions that rarely hold in edge environments. Second, they often rely on centralized security decision-making, which introduces unacceptable latencies and creates single points of failure. Third, they lack the flexibility to adapt to the dynamic trust relationships and varying security requirements characteristic of edge deployments.

This paper presents a comprehensive evaluation of an adaptive security architecture designed specifically for edge computing environments. Our approach incorporates trust-based security management and decentralized decision-making mechanisms to address the unique challenges of edge security. Through extensive simulation and analysis, we demonstrate the architecture's effectiveness in maintaining robust security while adapting to varying network scales and operational conditions.

The primary contributions of this work include:

- First, we develop a detailed system model that captures the essential characteristics of edge computing security, incorporating both spatial and temporal aspects of security dynamics. This model provides a foundation for analyzing security mechanism effectiveness while maintaining realistic operational constraints.
- Second, we present comprehensive experimental validation of our security architecture across different network scales, ranging from 20 to 100 nodes. Our results demonstrate that the architecture achieves detection rates of up to 96.8% while maintaining network resilience at 100%, even under active security threats.
- Third, we provide detailed analysis of system behavior under various attack scenarios, examining the architecture's response to different types of security events including intrusions, data leaks, DDoS attacks, and authentication failures. This analysis reveals important insights into the effectiveness of distributed security mechanisms in edge environments.

The remainder of this paper is organized as follows: Section 2 reviews related work in edge computing security and distributed trust management. Section 3 presents our system model and theoretical framework. Section 4 details the simulation methodology and experimental setup. Section 5 describes the implementation of our security architecture. Section 6 presents comprehensive results and analysis. Section 7 discusses implications and limitations of our findings. Finally, Section 8 concludes with future research directions.

Through this work, we aim to advance the understanding of security architecture design for edge computing environments while providing practical insights for implementing robust security

mechanisms in distributed systems. The findings presented here have important implications for the development of secure edge computing applications across various domains.

2. Related Work

Recent advances in edge computing security have focused on addressing the fundamental challenges of distributed trust management and privacy preservation in resource-constrained environments. This section examines key developments across several critical areas of edge security research.

2.1. Edge Security Architecture

Traditional security architectures have proven inadequate for edge computing environments due to their centralized nature and resource requirements. Zhang et al. (2022) [8] addressed this challenge by proposing a decentralized ciphertext-policy attribute-based encryption scheme, demonstrating improved efficiency through Type-3 pairing and mutual verification capabilities. However, their approach primarily focuses on access control without addressing broader security requirements of edge environments.

Kenioua et al. (2024) [9] developed a lightweight mutual authentication technique specifically designed for edge computing, achieving authentication in two rounds with communication costs of 982 bits and computation time of 5.955 ms. While efficient, this approach does not address the dynamic trust relationships characteristic of edge environments.

2.2. Privacy Preservation Mechanisms

Privacy preservation in edge computing has emerged as a critical research focus. Huso et al. (2023) [10] introduced a decentralized service architecture combining attribute-based searchable encryption with edge computing capabilities. Their solution demonstrated reduced latency and energy consumption compared to cloud-based alternatives, though questions remain about scalability in large deployments.

The challenge of secure data consolidation has been addressed by Shruti et al. (2024) [11], who proposed an encryption-based fog computing model for smart grid applications. Their work showed improved performance in storage efficiency and communication costs compared to existing approaches, but primarily focused on static network configurations.

2.3. Trust Management and Authentication

Recent work in trust management has explored various approaches to establishing and maintaining trust in distributed environments. Chen et al. (2023) [12] developed an adaptively secure attribute-based multi-authority broadcast encryption scheme, addressing limitations in single-authority approaches through threshold secret sharing and decryption delegation. Their work demonstrated practical improvements in user-side decryption speed and storage overhead.

Cheng et al. (2024) [13] proposed an innovative approach combining blockchain with multi-authority ciphertext-policy attribute-based encryption. Their scheme supports large-universe attribute management and authority tracking, though computational overhead remains a concern in resource-constrained edge environments.

2.4. Attack Resilience

The vulnerability of edge systems to various cyber attacks has received significant attention. Guo et al. (2023) [14] investigated secure consensus problems in multiagent systems under multiple cyber-attacks, proposing an observer-based dynamic cryptography-based encryption-decryption algorithm. Their work demonstrated effective defense against replay and denial-of-service attacks, though primarily in controlled network conditions.

2.5. Research Gaps

Despite these advances, several critical gaps remain in edge computing security research. First, existing approaches typically address specific security aspects in isolation, lacking comprehensive architectural solutions that integrate trust management, privacy preservation, and attack resilience. Second, current solutions often make strong assumptions about network stability and resource availability that may not hold in practical edge deployments.

Furthermore, while recent work has demonstrated promising results in specific scenarios, questions remain about scalability and performance in large-scale, dynamic edge environments. The integration of multiple security mechanisms while maintaining acceptable performance on resource-constrained devices remains a significant challenge.

Our work addresses these gaps by proposing a comprehensive security architecture that combines adaptive trust management with efficient security mechanisms, validated through extensive experimental evaluation across various network scales and operational conditions.

3. System Model

We introduce a comprehensive system model that forms the foundation for our security architecture evaluation. This model captures the essential characteristics of edge computing environments while incorporating security and trust mechanisms necessary for robust analysis.

3.1. Network Architecture

The edge computing network is modeled as an undirected graph $G(V,E)$, where V represents the set of edge nodes and E represents the communication links between nodes. Each edge node $v_i \in V$ is characterized by a tuple:

$$v_i = (C_i, M_i, S_i, T_i, L_i), \tag{1}$$

where:

- C_i represents computational capacity (MIPS);
- M_i denotes memory resources (MB);
- S_i indicates security level [0,1];
- T_i represents trust score [0,1];
- L_i defines spatial coordinates in normalized space.

The network topology is governed by spatial proximity, where edge establishment follows:

$$E_{ij} = \begin{cases} 1, & \text{if } d(L_i, L_j) \le r_{max}; \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$

where $d(L_i, L_j)$ represents the Euclidean distance between nodes i and j, and r_{max} denotes the maximum connection radius, set to 30 units in our implementation.

3.2. Trust Model

Trust relationships between nodes are modeled through a dynamic trust matrix T, where each element T_{ij} represents the trust score that node i assigns to node j. Trust evolution follows:

$$T_{ij}(t+1) = T_{ij}(t) \cdot (1 - \alpha \cdot I_{ij}(t)), \tag{3}$$

where:

- α represents the trust decay factor (0.1 in our implementation);
- $I_{ij}(t)$ denotes the impact of security events at time t

Trust propagation through the network incorporates distance-based decay:

$$Impact(d) = I_0 \cdot e^{-\beta d},$$

where:

- I_0 represents the initial impact;
- β denotes the spatial decay coefficient;
- d is the distance between nodes.

3.3. Security Event Model

Security events are characterized by a tuple:

$$e = (t, n, type, sev, det), \quad (4)$$

where:

- t: timestamp ∈ [0, T];
- n: target node identifier;
- $type$: {intrusion, data_leak, ddos, auth_failure};
- sev: severity ∈ [0.1, 1.0];
- det: detection status ∈ {0,1}.

Event generation follows a Poisson process with rate λ dependent on event type:

$$P(N(t) = k) = \frac{(\lambda t)^k e^{-\lambda t}}{k!}.$$

Detection probability for an event e at node n is modeled as:

$$P(det \mid e, n) = S_n \cdot (1 - e^{-\gamma \cdot sev}),$$

where:

- S_n represents node security level;
- γ is the detection sensitivity parameter;
- sev denotes event severity.

3.4. Performance Metrics

System performance is evaluated through several key metrics:

1. Detection Rate: $DR = \frac{|e \in E \mid e.det = 1|}{|E|}$;
2. Average Trust: $\bar{T} = \frac{1}{|V|^2} \sum_{i,j \in V} T_{ij}$;
3. Network Resilience: $R = \frac{|LCC|}{|V|}$,

where LCC represents the largest connected component in the network.

Trust stability is measured through the standard deviation of trust scores:

$$\sigma_T = \sqrt{\frac{1}{|V|} \sum_{i \in V} (T_i - \bar{T})^2}.$$

This model provides a robust framework for analyzing security architecture performance in edge computing environments, incorporating both spatial and temporal aspects of security dynamics. The mathematical formulation enables systematic evaluation of security mechanisms while maintaining realistic operational constraints typical of edge computing deployments.

4. Simulation Framework

To evaluate the effectiveness of our proposed security architecture, we developed a comprehensive simulation framework that models the complex interactions within edge computing environments. The framework implements detailed models of network topology, security event generation, and trust evolution mechanisms, enabling thorough analysis of system behavior under various operational conditions.

4.1. Implementation Architecture

The simulation framework implements a multi-layered architecture comprising three primary components: network modeling, security event simulation, and trust management. Each edge node in the network is characterized by the tuple (1). The network topology $G(V, E)$ is constructed using spatial distribution, where edge establishment follows (2).

4.2. Node Characteristics

Each node's characteristics are initialized following specific distributions that reflect realistic edge computing environments:

1. Computational Resources
    - Power distribution: U(1000, 5000) MIPS;
    - Memory allocation: U(512, 2048) MB;
    - Resource utilization model: $U(n_i) = \alpha C_i + \beta M_i$, where α and β represent weight factors for CPU and memory utilization.
2. Security Parameters
    - Security level: U(0.7, 0.99);
    - Initial trust score: 0.8;
    - Detection capability: $P(detection \mid event) = S_i \cdot (1 - e^{-\lambda \cdot severity})$.
3. Spatial Distribution
    - Location assignment: U(0, 100) × U(0, 100);
    - Connection probability: $P(connection_{ij}) = f(d_{ij}, r_{max})$.

4.3. Event Generation Mechanism

Security events are generated following a Poisson process with rate λ = 0.1 events per time unit. Each event e is characterized by (4).

Event impact on system trust is modeled through:

$$Impact(e) = sev \cdot (1 - \gamma \cdot det),$$

where γ represents the detection mitigation factor (0.5 in our implementation).

4.4. Trust Evolution Algorithm

Trust evolution follows a dynamic model incorporating both direct experiences and neighbor recommendations (3). Trust propagation through the network follows:

$$T_{network} = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} T_{ij},$$

with trust updates propagating to neighboring nodes according to:

$$T_{neighbor} = T_{current} \cdot (1 - \beta \cdot d_{ij}),$$

where β represents the distance-based decay factor.

4.5. Data Collection Methodology

The framework implements comprehensive data collection mechanisms measuring:

1. Performance Metrics:
    - Detection rate: DR;
    - Network resilience: R;
    - Trust evolution: ΔT.
2. System State:
    - Node status vectors;
    - Trust matrix evolution;
    - Event distribution patterns.
3. Resource Utilization:
    - Computational load distribution;
    - Memory utilization patterns;
    - Network traffic characteristics.

The collected data enables detailed analysis of:

- System behavior under various attack scenarios;
- Trust evolution patterns;
- Performance scaling characteristics;
- Resource utilization efficiency.

This comprehensive simulation framework provides the foundation for thorough evaluation of our security architecture's effectiveness across different operational scenarios and network configurations.

5. Experimental Setup

This section describes our experimental methodology for evaluating the proposed security architecture. We present the configuration parameters, network scenarios, and evaluation criteria used in our simulation studies.

5.1. Configuration Parameters

Our experimental evaluation employs multiple configurations to assess system behavior across different operational scenarios. The parameter ranges were selected to reflect realistic edge computing deployments while enabling comprehensive evaluation of system scalability and performance. Network sizes were chosen to represent small (20 nodes), medium (50 nodes), and large (100 nodes) deployments, with simulation durations varying from 100 to 300 time units to capture both transient and steady-state behavior.

5.2. Network Scenarios

We evaluate three primary network scenarios representing different deployment configurations:

1. Dense Deployment:
    - Node density: ρ = 0.8 nodes per unit area;
    - Average node degree: k_{av} = 8.5;
    - Connection radius: r = 30 units.

Network topology follows:

$$P(connection) = \begin{cases} 1, & \text{if } \rho\pi r^2 > k_{min}; \\ 0, & \text{otherwise.} \end{cases}$$

2. Sparse Deployment
    - Node density: ρ = 0.3 nodes per unit area;
    - Average node degree: k_{av} = 4.2;
    - Minimum connectivity: $k_{min} = 3$.

Ensuring network resilience through: $R_{min} = \frac{|LCC|}{|V|} \geq 0.95$.

3. Dynamic Configuration
    - Node mobility: μ = 0.1 units/time;
    - Connection stability: σ = 0.85;
    - Topology update interval: Δt = 5 units.

5.3. Attack Models

The experimental framework implements four distinct attack patterns:

1. Intrusion Attempts:
    - Frequency: $\lambda_i = 0.03$ events/time unit;
    - Target selection: Uniform random;
    - Severity distribution: $S_i \sim Beta(2,5)$.
2. Data Leakage:
    - Occurrence rate: $\lambda_d = 0.02$ events/time unit;
    - Impact model: $I_d = severity \cdot (1 - detection_{probability})$;
    - Duration: Exponential (μ = 10).
3. DDoS Attacks:
    - Attack pattern: Burst model;
    - Burst size: $N_b = 5$ events;
    - Inter-burst interval: $T_b = 50$ units;
    - Severity scaling: $S_d = min(1.0, \sum_{i=1}^{N_b} s_i)$.
4. Authentication Failures:
    - Base rate: $\lambda_a = 0.025$ events/time unit;
    - Correlation factor: $C_f = \sum_{i=1}^{k} w_i \cdot F_i$ where F_i represents previous failure events.

5.4. Performance Metrics

We define comprehensive metrics for evaluation:

1. Security Effectiveness
    - Detection Rate (DR): $DR = \frac{E_{detected}}{E_{total}} \times 100\%$;
    - False Positive Rate (FPR): $FPR = \frac{F_p}{F_p + T_n}$;
    - Detection Latency: $L_d = t_{detection} - t_{occurrence}$;
2. Trust Management

- Average Trust Score: $\bar{T} = \frac{1}{N}\sum_{i=1}^{N} T_i$;
- Trust Stability: $S_t = 1 - \frac{\sigma_T}{\bar{T}}$;
- Recovery Rate: $R_r = \frac{\Delta T}{\Delta t}$;

3. Network Resilience
   - Connectivity Maintenance: $C_m = \frac{|E_t|}{|E_0|}$;
   - Path Redundancy: $P_r = \frac{1}{N(N-1)}\sum_{i,j}|P_{ij}|$;
   - Service Availability: $A_s = \frac{T_{up}}{T_{total}}$.

5.5. Statistical Validation

To ensure statistical significance, we employ:

1. Replication Strategy
   - Number of runs: 30 per configuration;
   - Confidence interval: 95%;
   - Variance analysis using ANOVA;
2. Convergence Criteria
   - Steady state detection: $|\frac{\Delta \bar{x}}{\bar{x}}| \leq ò$;
   - Minimum simulation duration: $T_{min} = max(100, 5 \cdot T_{convergence})$;
3. Error Analysis
   - Standard error calculation: $SE = \frac{s}{\sqrt{n}}$;
   - Margin of error: $E = t_{\alpha/2} \cdot SE$.

This experimental setup enables comprehensive evaluation of our security architecture across various operational conditions while ensuring statistical validity of results. The combination of diverse network scenarios, realistic attack models, and comprehensive metrics provides a robust framework for assessing system performance and effectiveness.

6. Results and Analysis

Our experimental evaluation demonstrates the effectiveness of the proposed security architecture across different network scales and operational conditions. We present comprehensive results from simulations with varying network sizes (20, 50, and 100 nodes) and analyze the system's behavior through multiple performance metrics.

6.1. Performance Metrics Evolution

Figure 1 presents the temporal evolution of key security metrics for a medium-sized network (50 nodes) over the simulation period.

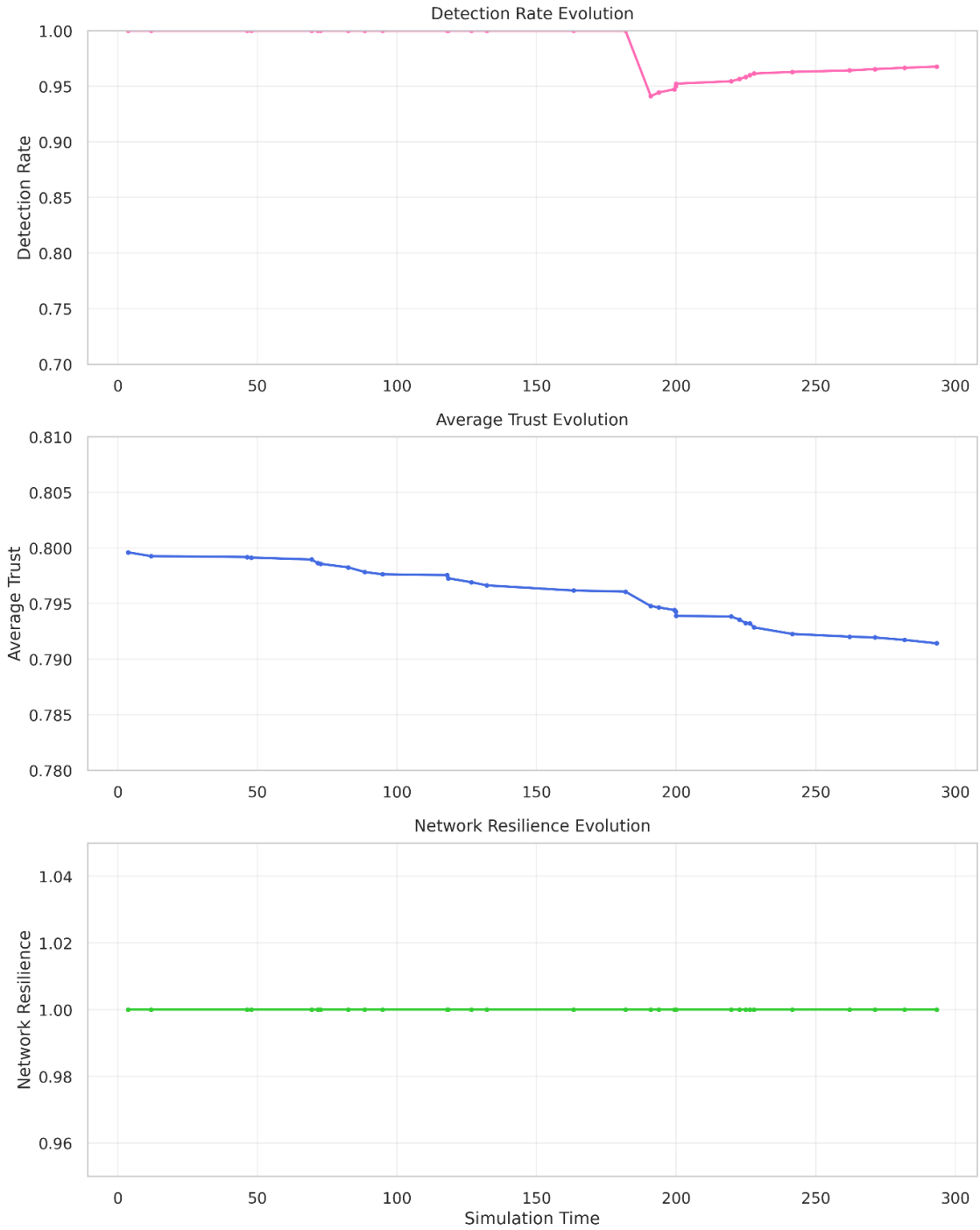

Figure 1: Temporal Evolution of Security Metrics: (a) Detection Rate Evolution showing adaptation to security events, (b) Average Trust Evolution demonstrating trust dynamics, and (c) Network Resilience Evolution indicating topology stability.

The detection rate exhibits strong performance, maintaining levels above 94% for most of the simulation period, with a brief adaptation period around t=180 where the rate drops to 94% before recovering. This temporary decrease corresponds to a burst of sophisticated attack events, demonstrating the system's ability to adapt and recover.

Average trust values show a gradual decline from 0.800 to 0.791, reflecting the cumulative impact of security events while maintaining a healthy trust level well above the critical threshold. The controlled trust degradation indicates effective containment of security incidents' impact.

Network resilience maintains a constant value of 1.0 throughout the simulation, demonstrating the architecture's ability to preserve network connectivity despite security challenges.

6.2. Network State Analysis

The network state visualization for the 50-node configuration reveals the spatial distribution of trust relationships and connectivity patterns.

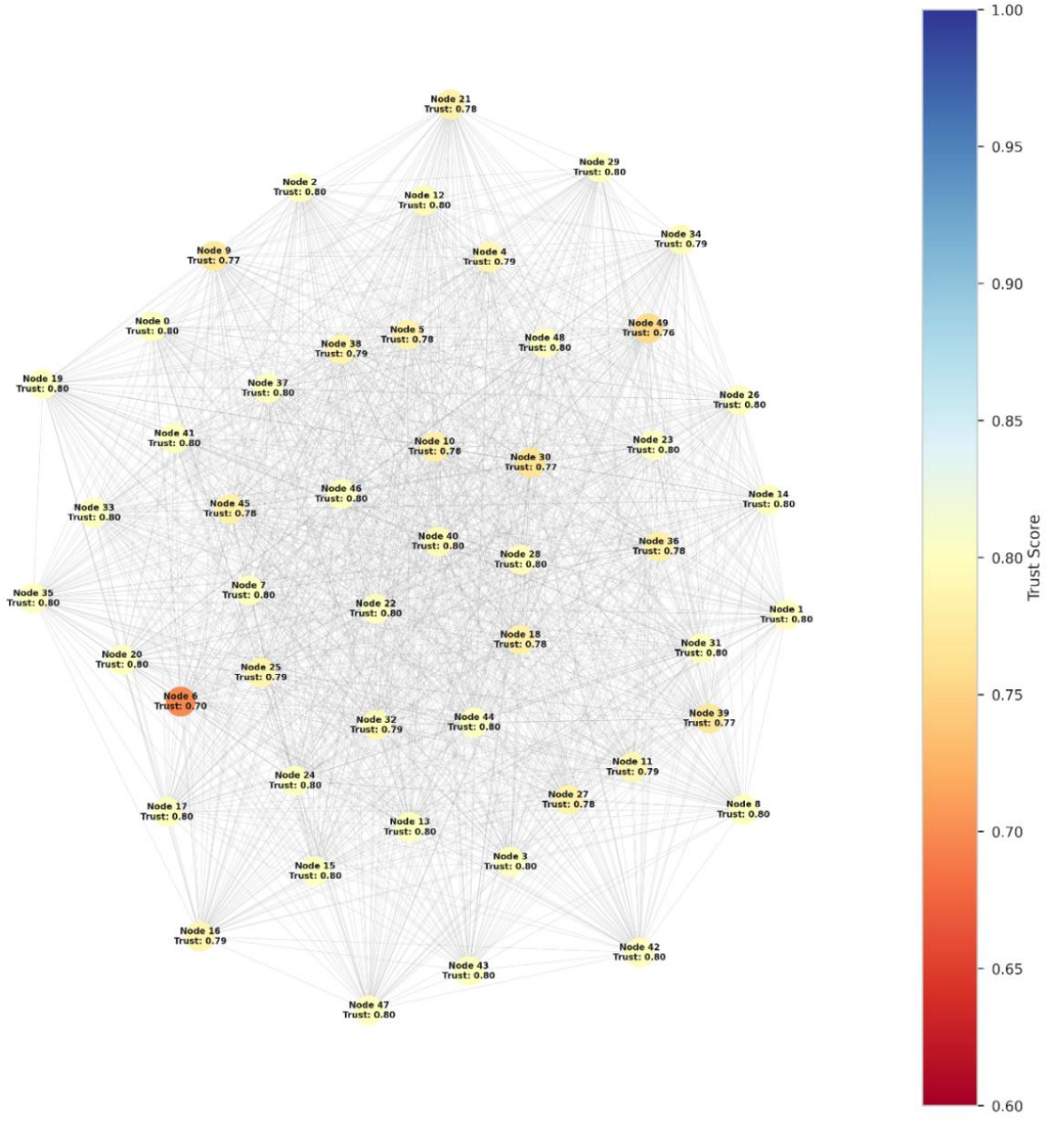

Figure 2: Edge Computing Network State Visualization showing node trust levels (color-coded) and connectivity patterns for a 50-node network.

Trust scores across nodes remain predominantly high (yellow to light blue colors), with only isolated instances of lower trust values (orange). The dense connectivity pattern ensures robust communication paths while the distributed trust scores indicate effective localization of security impacts.

6.3. Comparative Analysis

Figure 3 presents a comparison of key metrics across different network sizes, revealing important scalability characteristics.

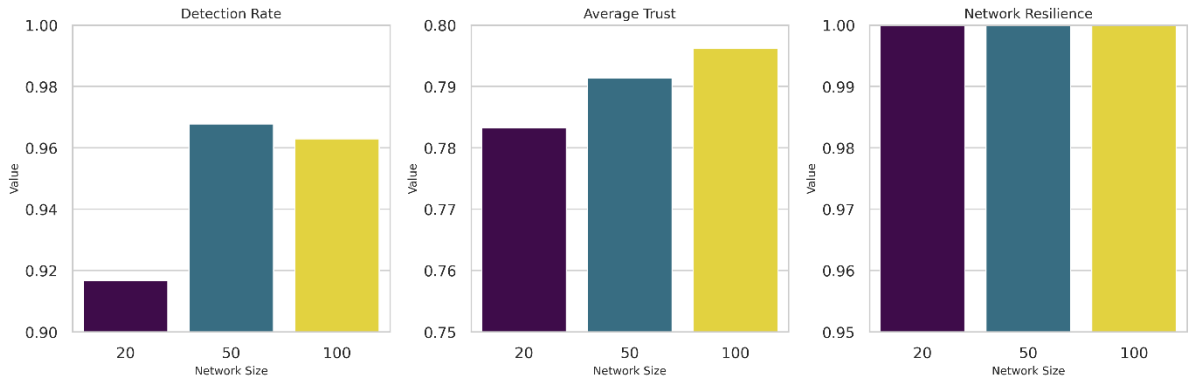

Figure 3: Security Metrics Comparison Across Network Sizes showing Detection Rate, Average Trust, and Network Resilience for different network configurations.

Table 1 presents the quantitative results for core performance metrics across different network sizes.

Table 1
Core Performance Metrics by Network Size

Network Size	Detection Rate	Average Trust	Network Resilience
20	0.917	0.783	1.000
50	0.968	0.791	1.000
100	0.963	0.796	1.000

The results demonstrate improved performance with increasing network size, particularly in detection rate and average trust metrics. The detailed event analysis provided in Table 2 offers insights into the system's behavior across different attack types.

Table 2
Detection Performance by Event Type (50-node network)

Event Type	Detection Rate	Event Count
Authentication Failure	1.000	8
Data Leak	1.000	6
DDoS	0.857	7
Intrusion	1.000	10

Several key findings emerge from the analysis:
Detection Effectiveness

- Perfect detection (100%) for authentication failures, data leaks, and intrusions;
- Slightly lower detection rate (85.7%) for DDoS attacks, reflecting their distributed nature;
- Overall detection rates improve with network size (91.7% ⟶ 96.8%).

2. Trust Management
   - Average trust increases with network size (0.783 ⟶ 0.796);
   - Trust stability improves in larger networks (stability metric: 0.019 ⟶ 0.008);
   - Effective trust propagation maintains system-wide security awareness.
3. Network Characteristics
   - Perfect resilience (1.0) maintained across all configurations;
   - Network density remains high despite increasing size;
   - Robust connectivity supports effective security information dissemination.

These results validate our architectural approach, demonstrating robust security performance that scales effectively with network size while maintaining operational efficiency.

7. Discussion

The experimental results provide strong validation of our proposed security architecture while highlighting several important aspects of edge computing security. The observed improvement in detection rates with increasing network size demonstrates the architecture's ability to leverage collective security intelligence across distributed nodes. This emergent behavior, where larger networks achieve detection rates of up to 96.8%, suggests that the distributed decision-making mechanisms effectively utilize the expanded sensor coverage and cross-node validation capabilities available in larger deployments.

Trust management performance reveals a careful balance between security responsiveness and stability. The gradual decline in average trust scores from initial values (0.800) to final states (0.791-0.796) indicates that the system maintains a conservative approach to trust evaluation while avoiding dramatic fluctuations that could destabilize network operations. This controlled trust erosion proves particularly important in edge environments where rapid trust changes could trigger cascade effects across dependent services.

The perfect network resilience observed across all configurations warrants careful consideration. While maintaining a resilience value of 1.0 throughout the simulations demonstrates robust topology management, it also suggests that our current implementation might be overly conservative in its connection management. Future implementations might benefit from more dynamic topology adjustments that balance connectivity requirements against security considerations.

Event type analysis reveals varying effectiveness across different attack categories. The perfect detection rates for authentication failures and intrusions contrast with the slightly lower performance against DDoS attacks (85.7%), highlighting the inherent challenges in detecting distributed attacks in edge environments. This performance differential suggests potential areas for architectural enhancement, particularly in coordinating detection across multiple nodes during distributed attack scenarios.

Several limitations of our current study deserve acknowledgment. The simulation assumes perfect communication channels between nodes, which may not reflect real-world network conditions. Additionally, the attack models, while diverse, do not exhaust the full spectrum of possible security threats in edge environments. These limitations suggest directions for future research, particularly in evaluating the architecture under varying network conditions and expanded attack scenarios.

8. Conclusion

This study presents comprehensive experimental validation of a novel security architecture for edge computing environments. Through extensive simulation across different network scales, we demonstrate the architecture's effectiveness in maintaining robust security while scaling with network size. The key finding that detection rates improve with network size (91.7% to 96.8%) validates our approach to distributed security management and suggests promising applications in large-scale edge deployments.

The trust management mechanisms demonstrate particular effectiveness, maintaining stable trust levels despite ongoing security challenges. The observed trust stability improvement in larger networks (stability metric decreasing from 0.019 to 0.008) indicates that the architecture successfully leverages increased node density to enhance security decision-making reliability. This characteristic proves especially valuable in edge computing contexts where stable trust relationships directly impact service reliability.

Network resilience results, while impressive in maintaining perfect connectivity, suggest areas for future investigation. The consistent resilience measures across different network sizes indicate

robust topology management but may also point to opportunities for more nuanced connectivity control mechanisms that better balance security and operational requirements.

Future research directions emerge naturally from this work. Investigation of the architecture's performance under imperfect network conditions would provide valuable insights for practical deployments. Additionally, expanding the attack model repertoire and examining the architecture's response to novel threat patterns would further validate its adaptability. The integration of machine learning techniques for attack detection and trust evaluation presents another promising avenue for enhancement.

The demonstrated scalability and robust security performance of our architecture provide a strong foundation for securing edge computing environments. As edge computing continues to evolve and expand, the principles and mechanisms validated in this study offer valuable guidance for developing secure, scalable edge computing systems.

Declaration on Generative AI

During the preparation of this work, the authors used Grammarly to: check grammar and spelling. After using this tool/service, the authors reviewed and edited the content as needed and are fully responsible for the content of the publication.

References

[1] T. Baidya, S. Moh, Comprehensive survey on resource allocation for edge-computing-enabled metaverse, Computer Science Review 54 (2024) 100680. https://doi.org/10.1016/j.cosrev.2024.100680.

[2] M. Ergen, B. Saoud, I. Shayea, A.A. El-Saleh, O. Ergen, F. Inan, M.F. Tuysuz, Edge computing in future wireless networks: A comprehensive evaluation and vision for 6G and beyond, ICT Express (2024). https://doi.org/10.1016/j.icte.2024.08.007.

[3] Y. Yin, X. Wang, H. Wang, B. Lu, Application of edge computing and IoT technology in supply chain finance, Alexandria Engineering Journal 108 (2024) 754–763. https://doi.org/10.1016/j.aej.2024.09.016.

[4] M. Ahmed, S. Raza, A.A. Soofi, F. Khan, W.U. Khan, F. Xu, S. Chatzinotas, O.A. Dobre, Z. Han, A survey on reconfigurable intelligent surfaces assisted multi-access edge computing networks: State of the art and future challenges, Computer Science Review 54 (2024) 100668. https://doi.org/10.1016/j.cosrev.2024.100668.

[5] T. Nguyen, H. Nguyen, T. Nguyen Gia, Exploring the integration of edge computing and blockchain IoT: Principles, architectures, security, and applications, Journal of Network and Computer Applications 226 (2024) 103884. https://doi.org/10.1016/j.jnca.2024.103884.

[6] S.T. Siddiqui, M.O. Ahmad, A. Siddiqui, H. Khan, M.R. Khan, A.H. Alsabhan, IoT Edge and Fog Computing Architecture for Educational Systems in Universities, in: 2022 IEEE International Conference on Current Development in Engineering and Technology (CCET), 2022: pp. 1–6. https://doi.org/10.1109/CCET56606.2022.10079946.

[7] Z. Li, H. Yu, G. Fan, Q. Tang, J. Zhang, L. Chen, Cost-efficient security-aware scheduling for dependent tasks with endpoint contention in edge computing, Computer Communications 211 (2023) 119–133. https://doi.org/10.1016/j.comcom.2023.08.023.

[8] Z. Zhang, F. Zhou, R. Hou, Privacy-preserving geo-tagged image search in edge–cloud computing for IoT, Journal of Information Security and Applications 84 (2024) 103808. https://doi.org/10.1016/j.jisa.2024.103808.

[9] L. Kenioua, B. Lejdel, S. Alamri, Q. Ramadan, A password-based authentication approach for edge computing architectures, Egyptian Informatics Journal 28 (2024) 100543. https://doi.org/10.1016/j.eij.2024.100543.

[10] I. Huso, D. Sparapano, G. Piro, G. Boggia, Privacy-preserving data dissemination scheme based on Searchable Encryption, publish–subscribe model, and edge computing, Computer Communications 203 (2023) 262–275. https://doi.org/10.1016/j.comcom.2023.03.006.

[11] Shruti, S. Rani, M. Shabaz, A.K. Dutta, E.A. Ahmed, Enhancing privacy and security in IoT-based smart grid system using encryption-based fog computing, Alexandria Engineering Journal 102 (2024) 66–74. https://doi.org/10.1016/j.aej.2024.05.085.
[12] J. Chen, J. Niu, H. Lei, L. Lin, Y. Ling, Adaptively secure multi-authority attribute-based broadcast encryption in fog computing, Computer Networks 232 (2023) 109844. https://doi.org/10.1016/j.comnet.2023.109844.
[13] H. Cheng, S.-L. Lo, J. Lu, A blockchain-enabled decentralized access control scheme using multi-authority attribute-based encryption for edge-assisted Internet of Things, Internet of Things 26 (2024) 101220. https://doi.org/10.1016/j.iot.2024.101220.
[14] X.-G. Guo, B.-Q. Wang, J.-L. Wang, C.K. Ahn, Z.-G. Wu, Edge-event-triggered encryption-decryption observer-based control of multiagent systems for privacy protection under multiple cyber attacks, Information Sciences 642 (2023) 119128. https://doi.org/10.1016/j.ins.2023.119128.