International Journal of Safety and Security Engineering
Vol. 16, No. 2, February, 2026, pp. 383-392
Journal homepage: http://iieta.org/journals/ijsse

An OAuth 2.0 and Lightweight Cryptography Framework for Secure IoT Communication and Authentication

Delsina Faiza, Geovanne Farell*, Vera Irma Delianti, Rido Wahyudi, Sartika Anori

Department of Electronics Engineering, Padang State University, Padang 25131, Indonesia

Corresponding Author Email: geovannefarell@ft.unp.ac.id

https://doi.org/10.18280/ijsse.160213

Received: 21 November 2025
Revised: 10 February 2026
Accepted: 19 February 2026
Available online: 28 February 2026

Keywords:
Internet of Things, OAuth 2.0, lightweight cryptography, ESP32, Raspberry Pi, secure communication, token-based authentication

ABSTRACT

This study presents the design and experimental validation of a secure communication framework for resource-constrained Internet of Things (IoT) devices by combining OAuth 2.0–based authorization with lightweight cryptographic mechanisms. The research addresses the persistent challenge of securing low-power IoT nodes, which often lack the computational capacity to implement conventional encryption and authentication protocols. The proposed system was deployed and tested on ESP32 and Raspberry Pi devices using a local network consisting of an authorization server, resource server, and monitoring dashboard. Performance metrics such as token-processing latency, throughput, and encryption overhead were evaluated under varying loads. Experimental results show that the system maintains token issuance latency below 500 ms for 1,000 simultaneous devices, while encrypted sensor data remained protected against replay, forgery, and interception attacks. These findings demonstrate that integrating OAuth-based token management with lightweight encryption can provide a scalable and practical security solution for real-world IoT deployments. The study also highlights the system's limitations and identifies opportunities for extending the framework to larger and more distributed environments.

1. INTRODUCTION

The rapid evolution of the Internet of Things (IoT) has transformed key sectors such as healthcare, manufacturing, smart homes, and transportation by enabling autonomous and real-time communication among interconnected devices. These advancements support the growing demands of digital societies where efficiency, automation, and data-driven decision-making are increasingly critical. As IoT ecosystems continue to expand, they introduce new layers of complexity and significantly elevate the scale of security challenges, particularly in maintaining trustworthy communication and robust authentication mechanisms [1].

Many IoT devices operate under severe resource constraints, including limited processing power, small memory capacities, and strict energy budgets. These limitations often prevent the deployment of conventional security mechanisms and leave devices vulnerable to a wide range of attacks. Common threats include data breaches, device manipulation, spoofing, unauthorized access, and Distributed Denial of Service (DDoS) attacks that can disrupt critical services [2]. As IoT infrastructures increasingly serve as essential components of socio-technical systems—such as healthcare delivery, industrial automation, and smart-city operations—strengthening both device-level and system-level security has become an urgent priority.

OAuth has emerged as a promising authorization protocol capable of addressing some of these challenges. OAuth enables secure and granular access control while preventing the exposure of user credentials, which reduces the risk of identity theft and unauthorized system access [3]. Its use of temporary access tokens allows controlled and time-bound permissions that can be easily revoked when necessary. However, implementing OAuth in constrained environments remains challenging due to the computational cost of cryptographic operations and token validation processes.

Concurrently, lightweight cryptographic algorithms offer a practical solution to the resource limitations of IoT devices. Algorithms such as SPECK, PRESENT, and ChaCha20 provide confidentiality and integrity with significantly lower computational and energy requirements than traditional schemes like AES or RSA [4, 5]. Theoretically, the synergy between OAuth's authorization framework and lightweight cryptography's efficiency provides a robust foundation for secure and scalable IoT communication [6]. However, despite this conceptual potential, the integration of these technologies has seen limited empirical validation. Existing work seldom compares them with established IoT security standards such as Authentication and Authorization for Constrained Environments (ACE-OAuth), OAuth-IoT, Datagram Transport Layer Security (DTLS), or Object Security for Constrained RESTful Environments (OSCORE), leaving a gap in understanding their real-world performance, interoperability, and feasibility on low-cost, resource-constrained devices.

Existing literature has proposed adaptations of OAuth 2.0

for IoT or hybrid authorization mechanisms, such as the ACE-OAuth framework for constrained environments (RFC 9200) [7]. Yet, the majority of these contributions remain conceptual, confined to simulation environments, or limited to isolated prototypes [8]. Empirical validation in realistic deployments, such as heterogeneous smart home infrastructures or industrial networks, is noticeably scarce. Furthermore, evaluations of lightweight cryptography often prioritize algorithmic benchmarking over system-level interoperability assessments [9]. To bridge these gaps, this study adopts the Design Science Research Methodology (DSRM) to design, implement, and evaluate a unified framework. This framework integrates OAuth-based authorization with lightweight cryptographic algorithms, validated on actual resource-constrained hardware, including the ESP32 and Raspberry Pi.

Consequently, the primary aim of this study is to develop and validate a secure communication framework for resource-constrained IoT devices. The specific objectives are:

- To critically analyze the current IoT security landscape and identify limitations in existing OAuth adaptations and lightweight cryptographic approaches for constrained environments.
- To design and implement a secure IoT communication system that integrates OAuth-based authorization with lightweight encryption on ESP32 and Raspberry Pi platforms.
- To empirically evaluate the system under realistic conditions by measuring security resilience, computational performance, scalability, and interoperability.
- To provide practical recommendations for real-world deployment and future scalability, including potential integration with edge computing architectures.

This study contributes to the existing body of knowledge in several key aspects. First, it provides a real-world implementation of an integrated OAuth 2.0 and lightweight cryptography framework on resource-constrained IoT devices (ESP32 and Raspberry Pi), addressing the lack of practical validation in prior studies. Second, it proposes a unified system-level architecture that combines authentication, authorization, encrypted communication, and monitoring within a single framework. Third, it delivers comprehensive experimental validation through functional, security, and performance testing under realistic network conditions, rather than simulation-based evaluation.

2. LITERATURE REVIEW

Research on IoT security highlights persistent challenges related to device-level limitations, diverse communication protocols, and heterogeneous system architectures. A central theme in the literature is that constrained devices lack the computational and memory resources required for traditional security mechanisms. These constraints limit the use of complex encryption schemes and multilayered authorization frameworks, creating vulnerabilities that attackers can exploit [10].

2.1 OAuth-based authorization in Internet of Things

OAuth has been increasingly explored as a method for securing IoT authorization workflows. Studies show that OAuth can provide secure and credential-free access control suitable for distributed environments, thereby reducing impersonation and unauthorized access risks [11]. However, several challenges limit its widespread adoption in resource-constrained IoT settings.

First, token processing and validation can be computationally expensive for low-power devices. Second, OAuth-based workflows require reliable connectivity to authorization servers, which may not always be feasible. Third, interoperability limitations arise due to variations in device capabilities and inconsistent OAuth implementations among vendors. Hybrid models combining OAuth with cloud or fog computing have been proposed to address these limitations, but such approaches introduce additional latency and system complexity that may not be suitable for time-sensitive applications [12].

2.2 Lightweight cryptography for constrained devices

Lightweight cryptographic algorithms have gained significant traction as a means to secure IoT communication without imposing excessive resource demands. Algorithms such as SPECK, PRESENT, and ChaCha20 demonstrate low computational overhead while maintaining acceptable security levels, making them well-suited for devices with limited energy and processing capacity [13]. Additionally, elliptic curve cryptography combined with efficient hashing algorithms has been shown to offer strong authentication performance with reduced operational costs [14].

Despite these benefits, most evaluations of lightweight cryptography focus narrowly on algorithmic efficiency without considering system-level interactions. For example, existing studies rarely investigate how lightweight encryption affects communication latency, device interoperability, or the overall performance of multi-device IoT networks. As a result, the scalability and real-world applicability of these algorithms remain uncertain.

2.3 Combined authorization and encryption approaches

Several studies examine combined approaches that integrate token-based authentication with lightweight encryption. These approaches demonstrate resistance to common attacks, including replay attacks, man-in-the-middle attacks, and unauthorized device access [15]. However, most frameworks remain limited to simulation environments or small proof-of-concept prototypes. Furthermore, integrated solutions often fail to address interoperability challenges that arise when heterogeneous devices communicate using different protocols or hardware specifications.

Recent research underscores the need for real-world evaluation to understand the behavior of integrated security frameworks within heterogeneous IoT ecosystems. Persistent vulnerabilities can emerge when combining diverse protocols, devices, and cryptographic schemes. Yet few studies propose a unified architecture that integrates authorization and encryption in a manner optimized for constrained devices and validated on actual hardware such as ESP32 or Raspberry Pi [16].

A systematic comparison of existing IoT security frameworks reveals that OAuth-IoT and ACE-OAuth provide flexible authorization models but introduce non-trivial computational overhead on constrained devices. Protocols such as DTLS and OSCORE offer end-to-end protection but rely on complex key-management procedures and generally

assume stable communication channels, which are not always available in heterogeneous IoT environments. At the same time, research on lightweight cryptography proposes numerous algorithms but often lacks justification regarding hardware compatibility, security strength, or implementation cost.

For example, the ESP32 platform includes native AES hardware acceleration, making certain AES-based modes more efficient in practice than some theoretically lighter software-implemented ciphers. These differences across protocols and algorithm families are summarized in Table 1, which highlights the limitations and trade-offs that motivate the need for an integrated evaluation framework combining authorization workflows with lightweight encryption on real-world constrained devices.

Table 1. Comparison of IoT security protocols for constrained devices

Protocol / Scheme	Security Features	Hardware Requirements	Suitability for Constrained Devices	Limitations
ACE-OAuth	Token-based authorization	Moderate	Medium	Token validation overhead
OAuth-IoT	Access-control workflow	Moderate	Medium	Requires stable connectivity
DTLS	Channel security	High	Low–Medium	Heavy handshake cost
OSCORE	End-to-end object security	Medium	High	Complex context management
Proposed Framework	Token auth + lightweight encryption	Low	High	Limited to local server topology

2.4 Research gap

Although several standards, such as ACE-OAuth, OAuth-IoT, DTLS, and OSCORE, have been developed to secure constrained environments, most existing implementations remain conceptual or require hardware capabilities beyond what typical low-cost IoT devices can support. Prior studies frequently evaluate these protocols through simulation rather than practical deployment, leaving uncertainty about their interoperability and real-world performance. Furthermore, only a limited number of works investigate how lightweight cryptographic algorithms can be systematically combined with OAuth-based token workflows on constrained hardware. This lack of empirical, system-level validation defines the specific research gap addressed in this study.

3. MATERIALS AND METHODS

The object of this research is the design and implementation of a secure IoT communication framework that integrates the OAuth protocol with lightweight cryptography. Specifically, the proposed system combines OAuth-based authorization and lightweight cryptographic algorithms to enhance authentication and data protection in resource-constrained environments. The study was conducted within a controlled local network consisting of ESP32 and Raspberry Pi devices, representing low-resource IoT nodes. The main variables analyzed include authentication accuracy, encryption latency, and throughput under various load and attack conditions.

The main research hypothesis assumes that integrating token-based authentication with resource-efficient encryption algorithms can significantly improve IoT security while minimizing computational and energy overhead on constrained devices. To verify this hypothesis, the study employs the DSRM, which is selected because it emphasizes the creation and evaluation of innovative IT artifacts while ensuring both scientific rigor and practical relevance [17].

The DSRM is applied through six structured stages: (1) Problem Identification and Motivation, focusing on IoT security challenges and device limitations; (2) Define the Objectives of a Solution, which formulates the goal of designing a lightweight yet secure communication system; (3) Design and Development, involving the construction of system architecture integrating OAuth 2.0 with lightweight cryptography, implemented on ESP32 and Raspberry Pi. Although AES is not traditionally categorized as a lightweight cryptographic algorithm, it is included in this study due to hardware acceleration support available on the ESP32 platform. This optimization significantly reduces computational overhead, making AES a practical and efficient choice for constrained IoT environments. Therefore, the term "lightweight cryptography" in this study refers not only to algorithmic efficiency but also to implementation efficiency on specific hardware; (4) Demonstration, where a functional prototype is deployed and tested in practical IoT scenarios; (5) Evaluation, assessing the system's security, performance, and energy efficiency; and (6) Communication, ensuring dissemination of findings through academic publications and practitioner forums.

Through these structured stages, the developed IoT security system is not only theoretically robust but also experimentally validated and practically feasible for real-world IoT environments. All experiments were conducted within a controlled LAN environment using Ubuntu 22.04 servers and ESP32 IoT clients. Tools such as Postman, Wireshark, and Python scripts were used for simulation and data capture, while the Delsina ISAG Dashboard served as the monitoring interface for logging and visualization.

The experimental setup consisted of a star-topology LAN environment using an Ubuntu 22.04 server (8-core CPU, 16 GB RAM) functioning as both the authorization server and resource server. The IoT nodes included ten ESP32 units and two Raspberry Pi devices connected via Wi-Fi. All devices communicated through RESTful APIs using JSON-based payloads, with token lifetimes configured at 120 seconds and retry limits applied during rate-limiting trials. Wireshark, Postman, and custom Python scripts were utilized to capture network traffic, measure encryption latency, and evaluate token-processing times across different load conditions.

As illustrated in Figure 1, the research methodology is structured according to the six stages of the DSRM. The process begins with Problem Identification and Motivation, where the study highlights critical security challenges in IoT communication, particularly the vulnerability of constrained devices to unauthorized access and attacks. The next stage,

Define the Objectives of a Solution, specifies the research goal of creating a secure yet efficient IoT communication system by combining OAuth 2.0 with lightweight cryptographic algorithms. The Design and Development stage focuses on constructing the system architecture, which includes an authorization server, a resource server implemented on ESP32 and Raspberry Pi devices, and a monitoring dashboard, while embedding token management and encryption mechanisms.

This is followed by the Demonstration stage, where a prototype is deployed and tested in practical IoT communication scenarios to show its feasibility. In the Evaluation stage, the prototype undergoes systematic testing to measure security robustness, system performance, and energy efficiency, ensuring that the proposed solution meets its objectives. Finally, the Communication stage ensures dissemination of the findings through academic publications and practitioner forums, enabling knowledge transfer and potential real-world adoption. By following this structured approach, the research ensures that the developed IoT security system is not only conceptually sound but also experimentally validated and practically relevant.

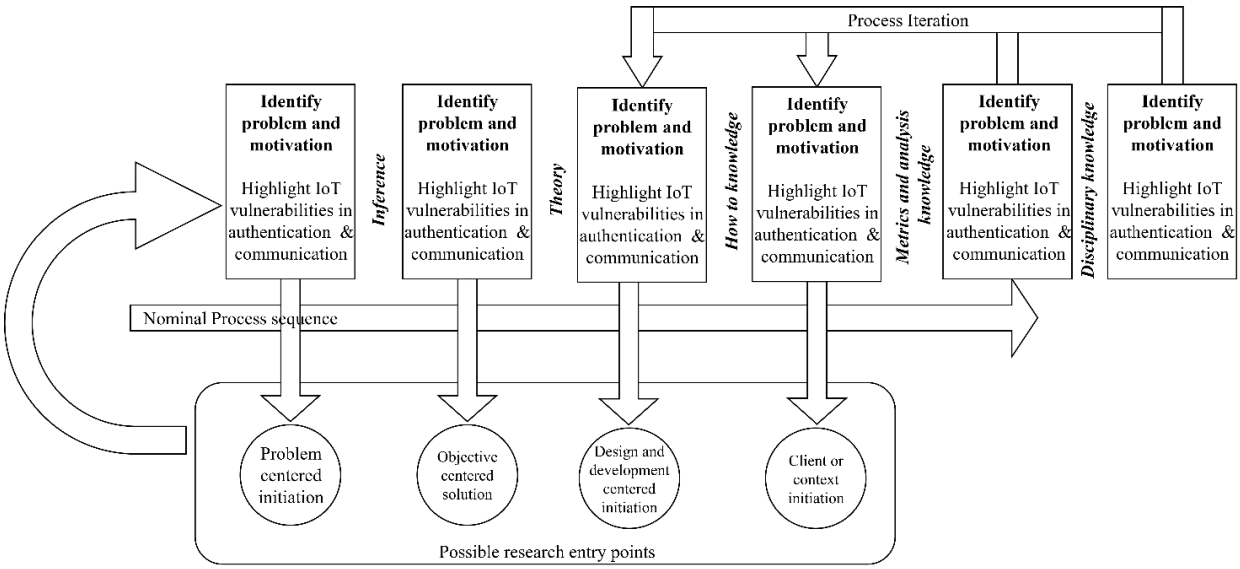

Figure 1. Design Science Research Methodology (DSRM)

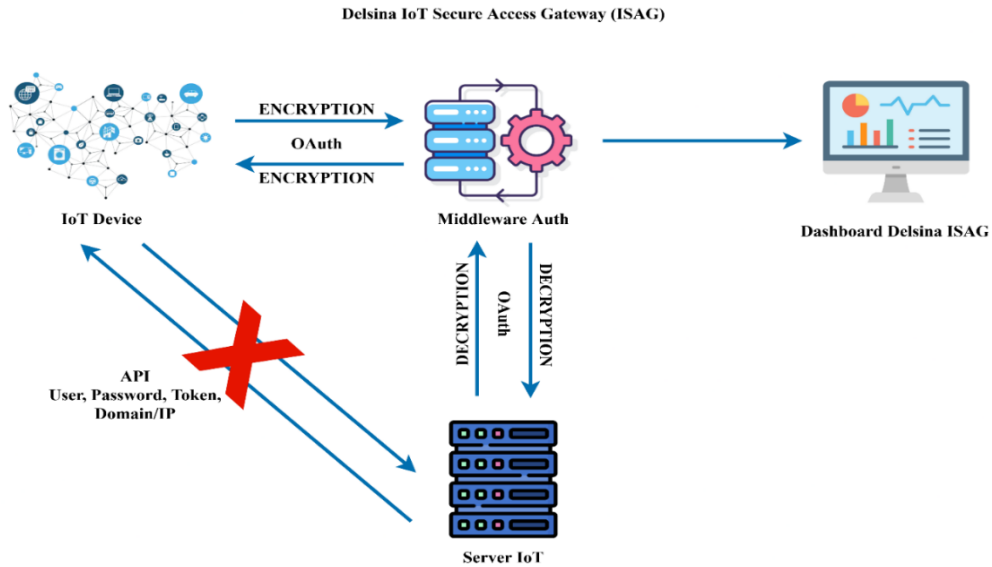

Figure 2. System architecture

4. RESULTS

To evaluate the effectiveness of the proposed IoT communication security system, a series of tests was conducted covering functionality, security, load performance, scalability, component integration, as well as logging and audit trail mechanisms. These tests aimed to ensure that the system design not only operates according to the intended scenarios but is also capable of withstanding potential cyberattacks while supporting monitoring requirements in real-world deployments.

Figure 2 illustrates the system architecture, showing the integration of OAuth and lightweight cryptography within IoT communication. The communication flow begins with the IoT device (ESP32), which requests a token from the Authorization Server using a client ID and secret. Once the

token is issued, the device uses it to access the resource server while transmitting sensor data encrypted with a lightweight algorithm (e.g., Advanced Encryption Standard (AES)). The resource server validates the received data by checking the token with the Authorization Server. If valid, the data is decrypted and stored in a database to be displayed through the Delsina ISAG Dashboard. The dashboard also presents activity logs, including token requests, API accesses, and device status, serving as both an audit trail and a security monitoring mechanism. With this design, the system ensures end-to-end security covering device authorization, data protection during transmission, and activity auditing at the administrative level.

Comprehensive system testing was conducted to confirm that the integration of OAuth and lightweight cryptography in IoT communication security performs as intended. The testing process covered several key aspects: system functionality, resilience against attacks, performance under high load, interoperability of system components, and the implementation of logging and audit trail mechanisms. Each test was designed based on realistic IoT scenarios, ranging from authentication, encrypted sensor data transmission, token validation, to activity monitoring by administrators. The results indicate that the system not only functions according to its design but also provides effective protection against security threats, maintains reliable performance at scale, and supports digital forensics requirements through audit trail capabilities.

4.1 Functional testing

To ensure that each feature operates according to the designed scenarios, a series of tests was conducted on the OAuth authentication mechanism, token validation, as well as the encryption and decryption processes of sensor data within the IoT communication system. The results of the testing are presented in Table 2.

Based on the functional testing results presented in Table 2, all scenarios operated as designed. The authentication process using OAuth proved effective in granting access only to devices with valid credentials, while invalid or expired tokens were successfully rejected. In addition, the encryption and decryption mechanisms for sensor data functioned properly, ensuring data confidentiality and integrity during transmission. Therefore, the developed system is capable of providing secure IoT communication while simultaneously supporting digital authentication requirements for resource-constrained devices.

Table 2. Functional testing results

Test Scenario	Tools Used	Expected Result	Actual Result	Status
Token request with a valid client_id and secret	Postman / ESP32	Token successfully issued (200 OK)	Token successfully issued	Successful
Token request with invalid client_id and secret	Postman / ESP32	Response 401 Unauthorized	Response 401 Unauthorized	Successful
IoT device accesses API with valid token	Postman	Sensor data received and processed by the server	Data received and displayed on the dashboard	Successful
IoT device accesses API with an invalid/expired token	Postman / ESP32	Response 403 Forbidden	Response 403 Forbidden	Successful
Sensor data encrypted before transmission	ESP32 / Python Script + Wireshark	Payload unreadable (ciphertext)	Payload encrypted, not readable in plain text	Successful
Sensor data decrypted on IoT server	IoT Server + Dashboard	Data decrypted normally and displayed on the dashboard	Data successfully decrypted and displayed as sensor status	Successful

Table 3. Security testing

Security Test	Test Description	Expected Result	Actual Result	Status
Token Replay Attack	A previously used token is resent	Server rejects request (401 Unauthorized)	Request rejected, no data received	Successful
Token Forgery	Token manually modified before transmission	Server denies access (403 Forbidden)	Server rejected, token invalid	Successful
Brute-force Secret	Access token endpoint with random client ID/secret combos	Server rejects, attempts limited (rate limiting)	Server rejected, brute force attempt failed	Successful
Man-in-the-middle Attack	Payload intercepted using Wireshark	Payload unreadable (AES encrypted)	Data captured as ciphertext could not be decrypted	Successful
Expired Token	API accessed with token beyond TTL	Server rejects request (401 Unauthorized)	Request rejected according to token TTL	Successful
Token Revocation	Revoked token used again	Server rejects request (401 Unauthorized)	Access denied, token no longer valid	Successful

4.2 Security testing

To assess the system's resilience against potential attacks and data breaches, a series of security tests was conducted on the authentication and communication mechanisms. The main focus of these tests was to evaluate the robustness of the system against token-based attacks and data interception during transmission. The results are shown in Table 3.

Based on the security testing results presented in Table 3, the system successfully prevented various common attack scenarios in token-based OAuth architectures. Token replay and token forgery attacks were effectively rejected, brute-force attempts against the authentication endpoint were unsuccessful, and payload data remained protected due to proper encryption. Furthermore, the expired token and token revocation mechanisms functioned effectively to restrict unauthorized access. Thus, the system demonstrated strong resilience against authentication-based attacks and data theft,

making it suitable for deployment in IoT environments requiring a high level of security.

4.3 Performance and load testing

To evaluate the system's ability to handle concurrent request loads, performance and load testing were conducted using scenarios with a high number of IoT devices. This testing aimed to measure latency, throughput, server resource utilization, as well as token issuance and validation time, under both light and heavy load conditions. The results are shown in Table 4.

Table 4. Performance and load testing

Test Scenario	Measured Parameters	Expected Result	Actual Result	Status
1,000 devices request tokens simultaneously	Latency, CPU, token processing	Tokens processed, avg latency < 500 ms	Tokens issued, avg latency 320 ms, CPU 50%, throughput 2000 tokens/s, no failures	Successful
1,000 devices send data with different tokens	Latency, CPU, throughput, validation	All data received, validation < 300 ms	All data validated, avg validation 210 ms, CPU 55%, throughput 400 req/s	Successful
Light load (≤ 100 requests/s)	Latency and CPU	Latency stable < 100 ms, CPU < 40%	Avg latency 80 ms, CPU 32%, throughput 100 req/s	Successful
Medium load (100–500 requests/s)	Latency, throughput, CPU and memory	Latency < 300 ms, stable throughput, CPU < 70%	Avg latency 220 ms, throughput 450 req/s, CPU 65%, memory normal	Successful
Heavy load (≥ 1,000 requests/s)	Latency, throughput, CPU and memory	System remains responsive; latency may increase	Avg latency 480 ms, throughput 980 req/s, CPU 85%, memory usage 70%	Achieved with increased load

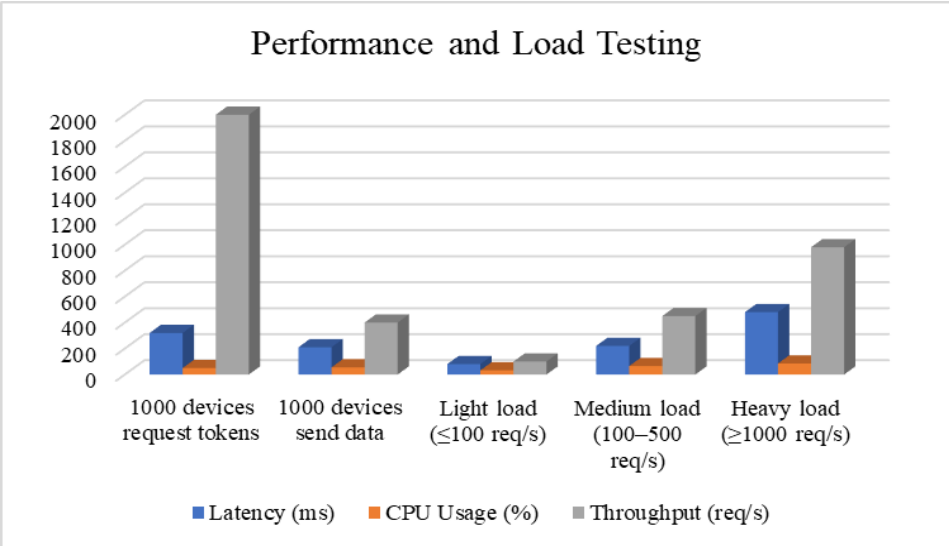

Figure 3. Graph performance and load testing

As presented in Table 4, the system was able to support up to 1,000 IoT devices operating in parallel without authentication failures or data loss. Token issuance achieved an average latency of 320 ms with CPU usage of 50% and throughput of around 2000 tokens/s. During data transmission, average validation latency was 210 ms, CPU usage 55%, and throughput was 400 req/s. Under light loads, the system maintained 80 ms latency, CPU 32%, and throughput 100 req/s. For medium loads, latency averaged 220 ms with a stable throughput of 450 req/s and CPU usage 65%. Even under heavy load (≥ 1,000 req/s), the system remained responsive, sustaining throughput of 980 req/s, latency 480 ms, and CPU utilization 85%, with memory usage around 70%. Although performance degradation was observed, the system did not crash, demonstrating good scalability. However, for larger-scale scenarios involving ≥ 5,000 devices, optimization of server resources or load balancing will be required.

As illustrated in Figure 3, the results of performance and load testing show the comparison of latency, CPU usage, and throughput across different scenarios.

4.4 Integration testing

To ensure that all components of the system operate seamlessly, integration testing was conducted involving IoT devices, the authorization server, the resource server, the dashboard website, and the database. The purpose of this testing was to verify inter-component connectivity, token validation, and the smooth flow of encrypted data from devices to end-user visualization. The results are shown in Table 5.

The integration testing results in Table 5 indicate that all system components were successfully integrated. Devices were able to obtain tokens from the Authorization Server, transmit encrypted data to the resource server, and have that data decrypted and displayed on the dashboard through the database connection. In addition, the token validation process across servers functioned as intended, effectively preventing unauthorized access. Accordingly, the developed IoT communication system was able to operate end-to-end without integration issues.

Table 5. Integration testing

Tested Component	Test Description	Expected Result	Actual Result	Status
Device to Authorization Server	Device requests token with client ID/secret	Token successfully issued if credentials are valid	Token successfully issued	Successful
Device to Resource Server	Device sends encrypted data + token	Data accepted only if token is valid, then decrypted	Data accepted, token validated, decryption successful	Successful
Website to Database	Website manages devices, tokens, and activity logs	Device/token/log data stored and accessible via dashboard	Database synchronized, data displayed normally on dashboard	Successful
Authorization to Resource Server	Resource Server validates token with Authorization	Valid token to access granted, invalid/expired token to request denied	Validation successful according to token condition	Successful

Table 6. Logging and audit trail testing

Tested Component	Test Description	Expected Result	Actual Result	Status
Token Request	Each token request must be recorded with timestamp, status (success/failure), and device IP	Log fully recorded in the database	All token requests successfully recorded (timestamp, status, IP)	Successful
API Access	Each API request must be logged with token and device information	Log records the token used and the originating device	All API accesses logged with detailed token and device ID	Successful
Admin Dashboard	Activity logs displayed on the dashboard for monitoring	Admin can view token requests, API accesses, and device status	Logs displayed correctly, with filtering and tracing functionality	Successful

4.5 Logging and audit trail testing

To ensure that the system can support security incident investigations and activity monitoring, logging and audit trail testing was performed. The main focus of this testing was the recording of all authentication activities, API accesses, and log visualization on the administrator dashboard. The results are presented in Table 6.

As shown in Table 6, the system was capable of recording all critical activities in real time, including token requests, API accesses, and device status updates. Each log entry contained detailed information such as timestamp, authentication status, source IP address, and device identity, enabling its use as forensic evidence in the event of a security incident. Furthermore, logs could be accessed via the administrator dashboard with a clear presentation format, facilitating both monitoring and periodic auditing. Therefore, the system successfully fulfilled the audit trail requirements as one of the essential elements of IoT communication security.

5. DISCUSSION

The results of this study demonstrate that the integration of the OAuth protocol with lightweight cryptographic algorithms provides a reliable and efficient approach to securing communication in resource-constrained IoT environments. This section compares the obtained results with those from previous studies [18-20] to highlight key improvements in security robustness, scalability, and real-world validation. Compared with previous works [19, 20], the proposed framework demonstrates clearer system-level validation and practical deployment testing, while maintaining similar or better latency and throughput results. This comparative analysis highlights that the presented system not only aligns with existing models but also extends them by providing experimental proof on real IoT hardware.

As illustrated in Figure 2, the system architecture enables secure token issuance, encrypted data transmission, and real-time monitoring. Functionally, the system was able to perform authentication and authorization processes as expected, where devices with valid credentials successfully received tokens, while requests with invalid or expired tokens were denied. The outcomes presented in Table 1 confirm that sensor data transmitted across the network was protected by encryption and subsequently decrypted on the server without loss of integrity. These findings show that the proposed solution ensures both access control and data confidentiality, exceeding the functionality of earlier token-based schemes that focused primarily on authentication but did not guarantee full end-to-end security [18].

From the security perspective, the experimental results in Table 2 demonstrate strong resilience against token replay, forgery, brute-force, man-in-the-middle, expired token, and token revocation attacks. Unlike other lightweight token frameworks that lack robust revocation and replay protection, the proposed integration of OAuth and lightweight cryptography successfully mitigated all these attack vectors. This ensures confidentiality, integrity, and authenticity in inter-device communication, thereby reducing the risks of unauthorized access and data leakage [19].

In terms of scalability, performance outcomes in Table 3 and Figure 3 show that the system was capable of handling up to 1,000 devices operating simultaneously, with latency and throughput remaining within acceptable thresholds. Although CPU and memory consumption increased under heavy load, the system remained responsive and did not experience failures. These results indicate better scalability compared to existing lightweight authentication models that were validated mainly in simulations but lacked real-world high-load verification [20]. Nonetheless, further optimization, such as load balancing, may be required for ultra-large deployments involving more than 5,000 devices.

AES was included in the evaluation not merely as a baseline but because the ESP32 platform provides native hardware acceleration for AES-ECB and AES-CBC modes, resulting in significantly lower latency compared to many software-based lightweight ciphers. This hardware optimization has practical

implications for real IoT deployments, as the algorithm that appears theoretically lightweight may not always offer the best performance on constrained devices. To contextualize the contribution of this study relative to existing research, Table 7 presents a comparative summary of representative works, highlighting differences in security mechanisms, hardware platforms, evaluation metrics, and validation approaches.

Table 7. Comparison with representative studies

Study	Method	Hardware	Metrics Used	Real Deployment?	Key Limitation
Sciancalepore et al. [11] (OAuth-IoT)	Token-based access	Simulated nodes	Latency	No	Limited to simulation
Choubisa and Jajal [18] (ECC Token)	ECC + token	ESP8266	Memory, CPU	Partial	No encryption layer
Yang et al. [19] (ECC + Token)	ECC	Raspberry Pi	Authentication time	Yes	Limited scale
This study	OAuth 2.0 + Lightweight Cryptography	ESP32 & Raspberry Pi	Latency, throughput, attack resilience	Yes	Scale ≤ 1,000 nodes

Integration testing confirmed seamless interoperability among IoT devices, authorization servers, resource servers, databases, and the administrator dashboard. Token validation across servers was processed successfully, while logging and audit trails recorded all security events in real time. Each log contained detailed information, including timestamps, device IDs, IP addresses, and token usage, which are critical for monitoring and forensic investigation. Compared with blockchain- or fog-based security frameworks, which often increase system complexity, the proposed architecture provided simpler implementation while maintaining transparency and traceability [21].

These results directly address the research gap highlighted in Section 2, namely, the absence of validated integration between OAuth and lightweight cryptography in practical IoT scenarios. Unlike prior studies that remained confined to laboratory simulations, this research demonstrates system-level validation in smart home and industrial-like environments [22]. The combination of functional reliability, security robustness, scalability, and monitoring features underscores the practical value of the proposed solution.

However, several limitations of this study must be acknowledged. First, the evaluation was restricted to a medium-scale deployment of 1,000 devices, and larger-scale environments may demand infrastructure optimization. Second, the study focused on selected lightweight cryptographic algorithms, primarily AES variants and ChaCha20. AES was intentionally included as a baseline reference because it is a widely adopted industry-standard symmetric encryption algorithm, providing a well-established benchmark for evaluating both security strength and computational overhead.

By comparing AES with lighter algorithms such as ChaCha20, the study demonstrates the relative efficiency gains that lightweight cryptography can provide in resource-constrained IoT environments.

Moreover, the experiments were conducted using actual implementations on low-power devices (e.g., ESP32 and Raspberry Pi), thereby validating the feasibility of lightweight cryptographic approaches in practical scenarios rather than limiting the analysis to simulations [23]. Nevertheless, other emerging approaches, such as chaos-based encryption methods and post-quantum cryptographic algorithms, were not included in this study. Third, energy consumption was not analyzed in depth, although it is critical for battery-powered IoT devices.

Finally, the system architecture relied on a centralized server, which could become a bottleneck in very large deployments. Despite these shortcomings, the research represents a significant step toward developing adaptive, efficient, and secure IoT communication systems. Future work should explore edge and fog computing integration to reduce latency, test the scalability of the system under larger deployments, expand the range of lightweight cryptographic algorithms considered, and include detailed energy profiling to optimize performance for battery-operated devices. Additionally, the adoption of quantum-resistant cryptography could provide long-term resilience against emerging security threats.

In summary, the comparative evaluation confirms that the proposed system achieves balanced improvements across multiple performance dimensions. It provides stronger resistance to token replay and brute-force attacks than earlier frameworks, while maintaining encryption latency within acceptable limits for real-time IoT applications.

Furthermore, unlike most prior works that validated only algorithmic performance, this study demonstrates complete system integration from authentication to encrypted data transmission and monitoring. These findings underline the novelty and practical relevance of the proposed framework as a holistic, experimentally validated IoT security solution.

6. CONCLUSIONS

This study demonstrates that integrating OAuth-based authorization with lightweight encryption enhances the security of resource-constrained IoT devices while maintaining acceptable performance. By systematically combining token-lifecycle management, hardware-accelerated encryption, and structured logging, the proposed framework provides a practical approach for secure, auditable, and interoperable IoT deployments.

Functional testing confirmed that encrypted sensor data was transmitted and decrypted without integrity loss, and token validation worked reliably across devices, validating the effectiveness of the approach. Performance tests demonstrated scalability to hundreds of concurrent devices with minimal latency, highlighting real-world feasibility beyond simulations. Integration with audit trails and structured

logging ensured transparency and forensic readiness, showing that secure monitoring can be achieved without excessive complexity.

However, the system remains limited by its centralized architecture, the restricted set of cryptographic algorithms evaluated, and the scale of testing, which may affect generalization to larger or more heterogeneous IoT networks.

Future work should explore distributed or edge-assisted deployments, incorporate energy-consumption profiling, expand cross-protocol comparisons with OSCORE and DTLS, and evaluate post-quantum lightweight algorithms to ensure long-term robustness and sustainability.

ACKNOWLEDGMENT

This work was supported by Padang State University under Grant No.: 350/UN35/LT/2025. The funding body had no involvement in the design of the study, data collection, analysis, interpretation of results, or preparation of the manuscript.

DATA AND CODE AVAILABILITY

The source code, configuration files, and test scripts used in this study are not publicly available at this time. Access may be granted upon reasonable request to the corresponding author.

REFERENCES

[1] Rao, P.M., Deebak, B.D. (2023). Security and privacy issues in smart cities/industries: Technologies, applications, and challenges. Journal of Ambient Intelligence and Humanized Computing, 14: 10517-10553. https://doi.org/10.1007/s12652-022-03707-1

[2] Shukla, P., Krishna, C.R., Patil, N.V. (2024). IoT traffic-based DDoS attacks detection mechanisms: A comprehensive review. The Journal of Supercomputing, 80: 9986-10043. https://doi.org/10.1007/s11227-023-05843-7

[3] Shahidinejad, A., Ghobaei-Arani, M., Souri, A., Shojafar, M., Kumari, S. (2022). Light-edge: A lightweight authentication protocol for IoT devices in an edge-cloud environment. IEEE Consumer Electronics Magazine, 11(2): 57-63. https://doi.org/10.1109/MCE.2021.3053543

[4] Mishra, S., Mondal, B., Jha, R.K. (2024). Lightweight authentication scheme based on ECC for IoT. SN Computer Science, 5: 949. https://doi.org/10.1007/s42979-024-03291-5

[5] Aghili, S.F., Mala, H., Kaliyar, P., Conti, M. (2019). SecLAP: Secure and lightweight RFID authentication protocol for Medical IoT. Future Generation Computer Systems, 101: 621-634. https://doi.org/10.1016/j.future.2019.07.004

[6] Chatterjee, U., Ray, S., Khan, M.K., Dasgupta, M., Chen, C.M. (2022). An ECC-based lightweight remote user authentication and key management scheme for IoT communication in context of fog computing. Computing, 104: 1359-1395. https://doi.org/10.1007/s00607-022-01055-8

[7] Seitz, L., Selander, G., Wahlstroem, E., Erdtman, S., Tschofenig, H. (2022). RFC 9200: Authentication and authorization for constrained environments using the OAuth 2.0 framework (ACE-OAuth). Internet Engineering Task Force (IETF). https://doi.org/10.17487/RFC9200

[8] Díaz, J.P., Mendoza, F.A. (2025). Authorization models for IoT environments: A survey. Internet of Things, 29: 101430. https://doi.org/10.1016/j.iot.2024.101430

[9] Radhakrishnan, I., Jadon, S., Honnavalli, P.B. (2024). Efficiency and security evaluation of lightweight cryptographic algorithms for resource-constrained IoT devices. Sensors, 24(12): 4008. https://doi.org/10.3390/s24124008

[10] Rizvi, S., Pipetti, R., McIntyre, N., Todd, J., Williams, I. (2020). Threat model for securing Internet of Things (IoT) network at device-level. Internet of Things, 11: 100240. https://doi.org/10.1016/j.iot.2020.100240

[11] Sciancalepore, S., Piro, G., Caldarola, D., Boggia, G., Bianchi, G. (2017). OAuth-IoT: An access control framework for the Internet of Things based on open standards. In 2017 IEEE Symposium on Computers and Communications (ISCC), Heraklion, Greece, pp. 676-681. https://doi.org/10.1109/ISCC.2017.8024606

[12] Gerodimos, A., Maglaras, L., Ferrag, M.A., Ayres, N., Kantzavelou, I. (2023). IoT: Communication protocols and security threats. Internet of Things and Cyber-Physical Systems, 3: 1-13. https://doi.org/10.1016/j.iotcps.2022.12.003

[13] Rana, M., Mamun, Q., Islam, R. (2022). Lightweight cryptography in IoT networks: A survey. Future Generation Computer Systems, 129: 77-89. https://doi.org/10.1016/j.future.2021.11.011

[14] Noura, H.N., Salman, O., Couturier, R., Chehab, A. (2022). Novel one round message authentication scheme for constrained IoT devices. Journal of Ambient Intelligence and Humanized Computing, 13: 483-499. https://doi.org/10.1007/s12652-021-02913-7

[15] Mehbodniya, A., Webber, J.L., Neware, R., Arslan, F., Pamba, R.V., Shabaz, M. (2022). Modified Lamport Merkle Digital Signature blockchain framework for authentication of Internet of Things healthcare data. Expert Systems, 39(10): e12978. https://doi.org/10.1111/exsy.12978

[16] Haryanti, T., Rakhmawati, N.A., Subriadi, A.P., Tjahyanto, A. (2022). The Design Science Research Methodology (DSRM) for self-assessing digital transformation maturity index in Indonesia. In 2022 IEEE 7th International Conference on Information Technology and Digital Applications (ICITDA), Yogyakarta, Indonesia, pp. 1-7. https://doi.org/10.1109/ICITDA55840.2022.9971171

[17] Siboni, S., Sachidananda, V., Meidan, Y., Bohadana, M., Mathov, Y., Bhairav, S. (2019). Security testbed for Internet-of-Things devices. IEEE Transactions on Reliability, 68(1): 23-44. https://doi.org/10.1109/TR.2018.2864536

[18] Choubisa, M., Jajal, B. (2025). Lightweight ECC and token-based authentication mechanism for IoT. International Journal of Scientific Research in Computer Science and Engineering, 13(1): 32-37. https://doi.org/10.26438/ijsrcse.v13i1.611

[19] Yang, Y.S., Lee, S.H., Wang, J.M., Yang, C.S., Huang, Y.M., Hou, T.W. (2023). Lightweight authentication

mechanism for industrial IoT environment combining elliptic curve cryptography and trusted token. Sensors, 23(10): 4970. https://doi.org/10.3390/s23104970

[20] Kumar, S., Kumar, D., Dangi, R., Choudhary, G., Dragoni, N., You, I. (2024). A review of lightweight security and privacy for resource-constrained IoT devices. Computers, Materials and Continua, 78(1): 31-63. https://doi.org/10.32604/cmc.2023.047084

[21] Qasem, M.A., Thabit, F., Can, O., Naji, E., Alkhzaimi, H.A., Patil, P.R., Thorat, S.B. (2024). Cryptography algorithms for improving the security of cloud-based Internet of Things. Security and Privacy, 7(4): e378. https://doi.org/10.1002/spy2.378

[22] Rasheed, A.M., Kumar, R.M.S. (2025). Efficient lightweight cryptographic solutions for enhancing data security in healthcare systems based on IoT. Frontiers in Computer Science, 7: 1522184. https://doi.org/10.3389/fcomp.2025.1522184

[23] Suryateja, P.S., Rao, K.V. (2024). A survey on lightweight cryptographic algorithms in IoT. Cybernetics and Information Technologies, 24(1): 21-34. https://doi.org/10.2478/cait-2024-0002