

Article

Edge-Assisted Timed Efficient Stream Loss-Tolerant Authentication over the Constrained Application Protocol (TESLA-CoAP) for Low-Latency and Scalable Sixth Generation (6G) Internet of Things (IoT) Networks

Eman Abouelkheir

Department of Computer Engineering, College of Computer, Qassim University, Buraydah 52571, Saudi Arabia; e.abouelkheir@qu.edu.sa

Abstract

The rapid deployment of sixth-generation (6G) Internet of Things (IoT) networks demands lightweight authentication mechanisms that provide low latency, high scalability, and robust security for resource-constrained devices operating in dynamic wireless environments. Conventional authentication approaches based on Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS), blockchain-assisted architectures, and Generic Bootstrapping Architecture (GBA)-based schemes introduce significant computational complexity, communication overhead, synchronization delays, and infrastructure dependencies, limiting their suitability for large-scale edge-assisted IoT environments. This paper proposes Lightweight Timed Efficient Stream Loss-Tolerant Authentication over the Constrained Application Protocol (L-TESLA-CoAP), a lightweight and infrastructure-independent authentication framework that integrates adaptive TESLA delayed-key authentication, CoAP communication, edge-assisted synchronization, replay-aware synchronization, SHA3-HMAC-based symmetric authentication, and rotating pseudonym identities to provide continuous packet-level authentication. The proposed framework was implemented and evaluated using a Python-based simulation environment under constrained 6G IoT communication scenarios with network sizes ranging from 50 to 1000 IoT devices. The comparative evaluation against CoAP, DTLS, TLS, Blockchain-CoAP, and GBA-Hybrid TESLA shows that the proposed framework achieves low authentication latency (approximately 0.8–1.3 s) and low energy consumption (approximately 60–75 mJ) while maintaining packet-loss recovery capability, reduced communication overhead, reduced computation time, low memory consumption, and authentication throughput. Furthermore, the proposed framework provides resilience against replay, packet injection, impersonation, synchronization manipulation, and denial-of-service attacks through adaptive synchronization and delayed key disclosure. These results indicate that L-TESLA-CoAP provides an efficient, scalable, and lightweight authentication solution suitable for next-generation edge-assisted 6G IoT applications.



Academic Editor: Jie Yang

Received: 15 May 2026

Revised: 7 July 2026

Accepted: 8 July 2026

Published: 17 July 2026

Copyright: © 2026 by the author.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\) license](https://creativecommons.org/licenses/by/4.0/).

Keywords: 6G IoT security; TESLA authentication; CoAP; lightweight authentication; adaptive synchronization; edge-assisted security; constrained IoT; data privacy

1. Introduction

The swift advancement of sixth-generation (6G) wireless communication systems is anticipated to accelerate the deployment of Internet of Things (IoT) ecosystems that enable

intelligent transportation systems, industrial automation, cyber-physical infrastructures, smart healthcare, metaverse applications, and edge-assisted autonomous services [1,2]. In contrast to earlier wireless generations, 6G networks are expected to deliver high connectivity density, ultra-low-latency communication, intelligent edge processing, and ongoing machine-type interaction among billions of IoT devices [1]. The integration of constrained IoT devices into open wireless communication environments poses cybersecurity and synchronization challenges, especially in dynamic edge network conditions marked by packet loss, intermittent connectivity, and resource constraints [3,4].

Constrained IoT devices generally function with limited processing power, memory capacity, low-energy batteries, and communication interfaces [3]. As a result, traditional Internet security frameworks designed for high-performance computing environments cannot be effectively implemented in large-scale IoT ecosystems. In real implementations, authentication overhead, synchronization latency, packet retransmission, and session management significantly impact communication efficiency, energy consumption, and network scalability [4,5]. Consequently, next-generation 6G IoT systems require lightweight authentication frameworks that maintain secure connectivity while minimizing computational and transmission overhead.

Recent advancements in edge-assisted IoT designs have moved security processing closer to end devices, reducing the burden on centralized authentication systems and improving synchronization responsiveness [6]. Edge-assisted security frameworks enhance scalability by dividing authentication and synchronization tasks among proximate edge gateways rather than relying solely on centralized infrastructure [6,7]. Nonetheless, a central gap remains: in restricted IoT contexts, lightweight continuous authentication systems are needed that can operate effectively in unstable wireless environments without imposing significant signaling overhead or synchronization complexity.

The Timed Efficient Stream Loss-Tolerant Authentication (TESLA) protocols have emerged as effective lightweight authentication methods that employ symmetric cryptographic operations and deferred key disclosure to facilitate scalable multicast authentication with minimal computational complexity [8,9]. TESLA-based authentication is especially appealing for limited IoT communication as it reduces costly asymmetric cryptographic processing while enhancing packet-loss tolerance and communication scalability [10,11]. Nonetheless, several current TESLA-oriented systems continue to depend on infrastructure-assisted synchronization, fixed synchronization intervals, or intricate initialization processes that constrain deployment adaptability in dynamic edge-assisted IoT contexts [12,13].

Concurrently, lightweight application-layer communication protocols such as the Constrained Application Protocol (CoAP) have gained widespread acceptance in IoT communication due to their reduced communication overhead, compact packet format, and suitability for constrained devices that use UDP [14,15]. While CoAP decreases communication complexity, ensuring safe continuous authentication over lightweight CoAP communication poses challenges in highly dynamic wireless environments, especially in ultra-dense 6G edge-assisted deployments.

This research introduces a Lightweight TESLA-CoAP (L-TESLA-CoAP) system for safe, continuous authentication in restricted 6G IoT environments, motivated by this central gap. The proposed framework combines lightweight CoAP communication with adaptive TESLA delayed-key authentication and edge-assisted synchronization management to deliver scalable and infrastructure-independent continuous authentication for limited IoT systems. In contrast to telecom-assisted authentication frameworks that rely on Bootstrapping Server Function (BSF) and Home Subscriber Server (HSS) infrastructures [16,17], the proposed framework operates autonomously, leveraging lightweight edge-assisted synchronization and symmetric cryptographic authentication. The proposed architecture

also incorporates adaptive synchronization control, rotating pseudonym identities, replay-aware synchronization management, and lightweight delayed verification to enhance authentication continuity in unstable wireless environments.

Although the aforementioned studies have advanced lightweight authentication for constrained Internet of Things (IoT) environments, several important limitations remain. The main research gaps are as follows:

- TLS- and DTLS-based authentication frameworks provide strong confidentiality and authentication guarantees; however, they require repeated handshakes, complex certificate management, high computational cost, and considerable communication overhead, making them unsuitable for large-scale, resource-constrained sixth-generation (6G) IoT deployments.
- Blockchain-assisted authentication frameworks improve decentralized trust management and data integrity but introduce synchronization latency, consensus overhead, storage requirements, and energy consumption, limiting their applicability to constrained IoT devices.
- Existing TESLA-based authentication protocols provide lightweight multicast authentication and packet-loss tolerance through delayed key disclosure; however, many still rely on fixed disclosure intervals, asymmetric initialization procedures, or infrastructure-assisted synchronization, limiting their adaptability to dynamic wireless network conditions.
- Generic Bootstrapping Architecture (GBA)-assisted TESLA frameworks reduce the dependence on asymmetric cryptography through telecom-assisted symmetric bootstrapping; nevertheless, they remain dependent on Bootstrapping Server Function (BSF), Home Subscriber Server (HSS), and SIM/eSIM infrastructures, limiting deployment flexibility in infrastructure-independent edge-assisted IoT environments.
- Most existing lightweight authentication frameworks do not simultaneously provide infrastructure-independent operation, adaptive synchronization, continuous packet-level authentication, replay-aware synchronization, privacy preservation, low computational complexity, and lightweight communication for ultra-dense edge-assisted 6G IoT networks.

Motivated by these research gaps, this paper proposes the Lightweight Timed Efficient Stream Loss-Tolerant Authentication over the Constrained Application Protocol (L-TESLA-CoAP) framework, which integrates adaptive TESLA delayed-key authentication, lightweight CoAP communication, edge-assisted synchronization, replay-aware synchronization, SHA3-HMAC-based symmetric authentication, and rotating pseudonym identities to provide scalable, lightweight, and secure authentication for constrained 6G IoT environments.

The subsequent sections of this work are structured as follows. Section 2 examines recent pertinent research on lightweight IoT authentication and synchronization frameworks. Section 3 presents an overview and design of the proposed L-TESLA-CoAP framework. Section 4 outlines the implementation environment and operational procedure. Section 5 details the synchronization phase and quantitative simulation analysis. Section 6 assesses the cybersecurity of the proposed framework. Ultimately, Section 7 summarizes the work and addresses prospective research avenues.

2. Related Work

Securing constrained Internet of Things (IoT) devices operating within fifth-generation (5G) and sixth-generation (6G) communication environments remains one of the most critical challenges in modern cyber-physical systems, edge computing, and massive machine-type communication (mMTC) infrastructures [1,2]. The growing number of heterogeneous

connected devices, combined with strict latency requirements, limited computational resources, intermittent connectivity, and dynamic edge-network conditions, has accelerated research into lightweight authentication and communication-security mechanisms optimized for constrained IoT deployments [3,4].

Conventional Internet security mechanisms, including Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS), were originally designed for resource-rich computing environments and therefore impose considerable computational and communication overhead when deployed on constrained IoT devices [5,6]. TLS relies heavily on public-key cryptography, certificate validation, and multi-stage handshake procedures to establish secure sessions between communicating entities [5]. Although DTLS was introduced as a UDP-compatible adaptation of TLS to support constrained wireless communication, it still suffers from significant packet fragmentation, retransmission overhead, session-establishment latency, and energy consumption in lossy IoT environments [6,18]. These limitations become increasingly problematic in large-scale 6G edge-assisted IoT ecosystems where ultra-low latency and energy efficiency are essential requirements.

To address constraints imposed by conventional Internet protocols, lightweight communication frameworks such as the Constrained Application Protocol (CoAP) have been widely adopted for constrained IoT environments [19,20]. CoAP is specifically designed for low-power, low-bandwidth devices operating over UDP, while maintaining lightweight REST-based communication capabilities [10]. Due to its minimal packet overhead and reduced processing requirements, CoAP has become one of the most suitable communication protocols for constrained IoT systems deployed in smart cities, industrial automation, healthcare IoT, and edge-assisted cyber-physical infrastructures [20]. Nevertheless, CoAP itself does not provide comprehensive security services and therefore commonly relies on DTLS or token-based authentication mechanisms to establish secure communication channels [19,21].

Although CoAP-DTLS architectures improve communication confidentiality and authentication compared with unsecured CoAP implementations, several important limitations remain unresolved. In [21], Garcia-Carrillo and Marin-Lopez proposed a lightweight CoAP-based bootstrapping mechanism that integrates the Generic Bootstrapping Architecture (GBA) with CoAP and DTLS to improve authentication efficiency for constrained IoT devices. The proposed framework successfully adapted GBA procedures to IoT-oriented communication environments while reducing some of the computational overhead associated with traditional TLS-based authentication. However, the architecture still depended on DTLS handshake operations and repeated credential exchanges, which may introduce latency and synchronization overhead unsuitable for ultra-massive IoT scenarios characterized by intermittent connectivity and lossy wireless communication [21].

Similarly, Raza et al. [19] proposed Lithe, a lightweight secure CoAP framework designed to optimize DTLS for constrained IoT communication. Their work reduced communication overhead compared with traditional TLS implementations while preserving secure communication properties. Nevertheless, the proposed architecture still relied on DTLS session establishment and pre-shared-key management, which may expose systems to replay attacks, increase session management complexity, and incur additional synchronization overhead under highly dynamic network conditions [19]. Chen and Kunz [20] further evaluated the performance of IoT communication protocols under constrained wireless networks and demonstrated that CoAP-based architectures provide improved efficiency compared with HTTP-based systems. However, packet loss, retransmissions, and authentication delays remained significant challenges, especially when DTLS-based security mechanisms were employed over unstable wireless channels [20].

Recent research efforts have also explored blockchain-assisted authentication architectures to improve trust management, certificate distribution, and decentralized security in IoT environments [8,9]. Garba et al. [8] proposed LightCert4IoTs, a lightweight, blockchain-based certificate management framework designed to reduce certificate overhead and improve decentralized authentication for IoT applications. The framework introduced Lightweight Elliptic Curve Qu-Vanstone (L-ECQV) implicit certificates to reduce message complexity and computational cost. Although the blockchain-based approach enhanced certificate transparency and decentralized trust management, it introduced additional latency, storage complexity, consensus overhead, and computational demands unsuitable for highly resource-constrained IoT devices [8].

Similarly, Khoury et al. [9] integrated blockchain technology with CoAP and DTLS to improve lightweight certificate management and decentralized trust verification for constrained IoT environments. The proposed framework reduced dependence on traditional centralized certificate authorities while introducing lightweight certificate validation procedures compatible with IoT devices. However, blockchain-assisted approaches remain computationally intensive and may increase energy consumption and communication overhead due to blockchain synchronization, certificate verification, and distributed consensus operations [9]. Scalability and latency remain open challenges in ultra-dense 6G IoT deployments involving thousands or millions of heterogeneous devices operating simultaneously.

In addition to blockchain-assisted architectures, several studies have investigated lightweight cryptographic frameworks specifically optimized for constrained IoT environments [12–17]. Dutta et al. [12] presented a comprehensive survey on lightweight cryptographic approaches for insecure IoT systems and highlighted the tradeoff between security strength and computational efficiency in constrained environments. Similarly, Taha et al. [13] analyzed secure communication approaches for resource-constrained IoT devices and emphasized the need for authentication mechanisms that reduce energy consumption and communication overhead without compromising cybersecurity. Other studies [16,17] investigated lightweight encryption frameworks and IoT communication protocols, identifying significant limitations in synchronization, scalability, replay protection, and continuous authentication.

Timed Efficient Stream Loss-Tolerant Authentication (TESLA) protocols have emerged as one of the most promising lightweight authentication mechanisms for constrained IoT environments due to their low computational complexity, delayed key-disclosure mechanism, multicast scalability, and packet-loss tolerance [22,23]. TESLA protocols rely primarily on symmetric cryptographic operations and one-way hash chains to authenticate transmitted packets, while significantly reducing the computational demands of asymmetric cryptography [22]. Consequently, TESLA-based architectures are highly suitable for resource-constrained IoT devices operating in lossy, latency-sensitive wireless environments [23].

Several TESLA variants have been proposed to improve authentication robustness, synchronization efficiency, and packet-loss recovery capability in constrained IoT systems. Eledlebi et al. [15] proposed Hybrid TLI- μ TESLA, which improved authentication continuity and synchronization efficiency by combining multiple hierarchical keychains with SHA-2 and the Secure Hash Algorithm 3 (SHA-3). Their analysis demonstrated low computational complexity and improved packet-loss recovery capability compared with conventional TESLA variants [9]. Furthermore, Eledlebi et al. [10] conducted a comparative analysis of TESLA variants and demonstrated that Hybrid TLI- μ TESLA achieved improved resilience against packet loss, replay attacks, and synchronization failure while maintaining low computational overhead suitable for constrained IoT environments.

Enhanced inf-TESLA [10] further improved continuous connectivity and lightweight authentication performance by minimizing synchronization overhead and supporting continuous packet authentication in dynamic IoT environments. Similarly, Al Dhaheri et al. [24] proposed a two-level μ TESLA protocol optimized for constrained IoT communication environments, improving synchronization efficiency and reducing authentication delay. Although these TESLA variants significantly improve authentication efficiency and loss tolerance, most TESLA architectures still rely on asymmetric cryptography during the synchronization phase to securely exchange commitment keys, disclosure intervals, and keychain parameters between communicating entities [25,26]. This remaining dependence leaves a critical gap in the quest for fully lightweight authentication in constrained IoT environments.

The reliance on asymmetric cryptography during synchronization imposes several important limitations on constrained IoT devices. Public-key operations, certificate validation, and digital signatures require substantial processing capability and energy consumption, which may significantly reduce network lifetime and device efficiency [27,28]. Moreover, public-key infrastructures remain vulnerable to certificate-management complexity, replay attacks, brute-force attacks, and man-in-the-middle (MITM) attacks in highly dynamic IoT environments [29,30]. Wu et al. [31] further emphasized that lightweight authentication frameworks for cloud-assisted IoT systems must minimize reliance on expensive public-key operations to improve scalability and reduce communication delay.

To address the limitations of public-key synchronization in TESLA architectures, Eledlebi et al. [32] proposed Bio-Integrated Hybrid TESLA, which replaced conventional public-key synchronization with biometric authentication and fully symmetric key establishment. Although the proposed framework improved privacy preservation and reduced computational overhead, biometric synchronization may introduce additional implementation complexity and hardware dependency unsuitable for certain constrained IoT environments [32].

Generic Bootstrapping Architecture (GBA) has also attracted considerable attention as a lightweight authentication framework for constrained mobile IoT environments [5,6,33]. GBA leverages existing 3GPP mobile-network infrastructure, including the Home Subscriber Server (HSS), Bootstrapping Server Function (BSF), and SIM/eSIM authentication mechanisms, to establish secure session keys between devices and application servers without transmitting sensitive credentials over the network [6]. Because cryptographic operations are partially offloaded to the mobile network infrastructure, GBA significantly reduces the computational burden on constrained IoT devices while preserving strong mutual authentication capabilities [5,6].

Several recent studies have explored integrating GBA with lightweight IoT communication protocols. Ericsson's GBA-CoAP framework [8] demonstrated the feasibility of combining GBA with CoAP communication for constrained IoT devices while reducing authentication complexity compared with traditional DTLS architectures. However, the proposed framework still relied on conventional secure session establishment mechanisms and lacked continuous, lightweight authentication capabilities [14].

More recently, Eledlebi et al. proposed integrating GBA with Hybrid TESLA to replace public-key cryptography during the TESLA synchronization phase [1]. Their proposed architecture leveraged mobile network infrastructure to establish symmetric session keys without transmitting sensitive credentials, thereby improving synchronization efficiency and reducing computational overhead. The proposed GBA-Hybrid TESLA protocol successfully demonstrated the feasibility of replacing public-key synchronization with telecom-assisted symmetric bootstrapping [1]. Nevertheless, despite its significant advantages, the framework remains dependent on telecom-specific infrastructure entities such as

BSF, HSS, and SIM/eSIM-based authentication procedures, which may limit deployment flexibility in decentralized edge-assisted IoT ecosystems and infrastructure-independent cyber-physical environments.

Mun et al. [33] proposed an edge-assisted lightweight authentication framework for ultra-massive 6G IoT environments to reduce centralized authentication overhead and improve scalability. Their work highlighted the importance of edge-assisted security management for constrained IoT systems. However, the proposed approach did not address adaptive TESLA delayed-key authentication, lightweight CoAP-based continuous authentication, rotating pseudonym privacy, or adaptive synchronization under packet-loss conditions.

Furthermore, most existing lightweight authentication approaches continue to focus primarily on initial authentication rather than continuous lightweight authentication suitable for ultra-massive 6G IoT communication. Existing architectures often neglect important requirements such as adaptive synchronization, dynamic trust management, rotating pseudonym privacy preservation, edge-assisted packet verification, and infrastructure-independent continuous authentication under intermittent edge-network conditions.

Recent studies have increasingly focused on developing lightweight, scalable authentication mechanisms for sixth-generation (6G) Internet of Things (IoT) environments while addressing ultra-low latency, edge-native computing, and emerging quantum threats. Turnip et al. [34] presented a comprehensive survey of authentication and key agreement protocols for 6G networks, highlighting the limitations of conventional 5G authentication and emphasizing the need for lightweight hybrid post-quantum authentication mechanisms suitable for resource-constrained IoT devices. Bensari et al. [35] proposed a lightweight dual-blockchain authentication framework for 6G-enabled IoT environments that improves decentralized trust management using Fog Radio Access Networks (Fog-RANs) and smart contracts; however, blockchain synchronization and consensus operations still introduce additional latency and computational overhead. Alnaim et al. [36] investigated Zero-Trust mechanisms for distributed edge and fog computing environments and demonstrated that continuous identity verification enhances security in 6G systems, though the proposed architecture incurs increased policy management complexity and computational requirements. Faleh et al. [37] introduced a lightweight quantum-resilient authentication architecture for mobile edge computing-enabled 6G IoT networks that incorporates revocable authentication and post-quantum cryptographic primitives to address future quantum threats; however, integrating post-quantum algorithms increases computational complexity for constrained devices. Furthermore, recent experimental evaluations of post-quantum cryptography for 6G systems have shown that although NIST-standardized post-quantum algorithms provide strong resistance to quantum attacks, their larger key sizes and higher computational requirements remain challenging for latency-sensitive, resource-constrained IoT deployments [38]. These recent studies demonstrate significant progress toward secure 6G IoT authentication; however, challenges related to infrastructure dependency, computational overhead, synchronization efficiency, communication latency, and lightweight continuous authentication remain open research problems. These limitations further motivate the proposed L-TESLA-CoAP framework, which provides infrastructure-independent operation, adaptive TESLA delayed-key authentication, lightweight CoAP communication, edge-assisted synchronization, and low-overhead symmetric authentication for constrained 6G IoT environments.

Recent advances in 6G communication technologies highlight key enablers for future secure IoT systems. Reconfigurable and movable antennas (RMA) improve coverage, efficiency, and sensing capabilities [39], while Free Space Optics (FSO)-assisted IoT enables high-capacity, low-latency, and reliable communication [40]. Although these works do

not focus on authentication, they provide a strong foundation for secure infrastructure, which the proposed L-TESLA-CoAP framework can extend to offer lightweight, adaptive authentication support.

Table 1 summarizes the comparison between existing lightweight authentication approaches and the proposed L-TESLA-CoAP framework.

Table 1. Comparative Summary of Existing Lightweight Authentication Frameworks and the Proposed L-TESLA-CoAP Scheme.

Category/Scheme	Representative References (Updated)	Authentication Mechanism	Infrastructure Dependency	Continuous Authentication	Lightweight Suitability	Scalability	Main Limitations
TLS/DTLS-Based IoT Security	[5–7,14]	TLS/DTLS handshake, Public Key Infrastructure (PKI), Pre-Shared Key (PSK)	Certificate authority/DTLS infrastructure	No	Moderate	Moderate	Handshake overhead, fragmentation, retransmissions, and high energy consumption.
Blockchain-Assisted IoT Authentication	[15,19,35]	Blockchain certificates, decentralized trust, smart contracts	Blockchain infrastructure	Partial	Low	Moderate	Consensus latency, distributed ledger synchronization, storage overhead, and high computational complexity.
Lightweight Cryptographic IoT Frameworks	[17,18,20–23, 26,28,29,38]	Lightweight symmetric/asymmetric cryptography and post-quantum cryptography	Centralized or distributed trust	No	Moderate–High	Moderate	Lack of continuous authentication, multicast scalability, and increased complexity of post-quantum algorithms.
TESLA-Based Lightweight Authentication	[8–13]	Delayed key disclosure and hash-chain authentication	Public-key synchronization or biometric initialization	Yes	High	High	Initial synchronization overhead, fixed disclosure intervals, and asymmetric initialization dependency.
GBA-Based IoT Authentication	[1,24,30–32]	GBA-assisted symmetric key establishment	Bootstrapping Server Function (BSF), Home Subscriber Server (HSS), SIM/eSIM telecom infrastructure	Partial	Moderate–High	High	Telecom infrastructure dependency and centralized trust architecture.
Edge-Assisted/Zero-Trust Lightweight Authentication	[33–37]	Edge-native authentication, Zero-Trust verification, adaptive authentication, lightweight post-quantum authentication	Edge infrastructure, MEC/Fog infrastructure	Partial–Yes	High	Very High	Policy management complexity, synchronization dependency, infrastructure requirements, and increased computational overhead for post-quantum security.
Proposed L-TESLA-CoAP	—	Adaptive TESLA + CoAP + edge-assisted lightweight synchronization	Infrastructure-independent	Yes	Very High	Very High	Requires loose time synchronization.

3. Overview and Design of the Proposed L-TESLA-Coap Framework

The overall architecture of the proposed framework is illustrated in Figure 1. The operational steps of the proposed L-TESLA-CoAP architecture are:

- Step 1: TESLA Keychain Provisioning and Device Initialization;
- Step 2: Lightweight Synchronization Request;
- Step 3: Edge-Assisted Synchronization Response;
- Step 4: Authenticated CoAP Packet Transmission;
- Step 5: Delayed TESLA Key Disclosure;
- Step 6: Authentication Confirmation and Interval Update.

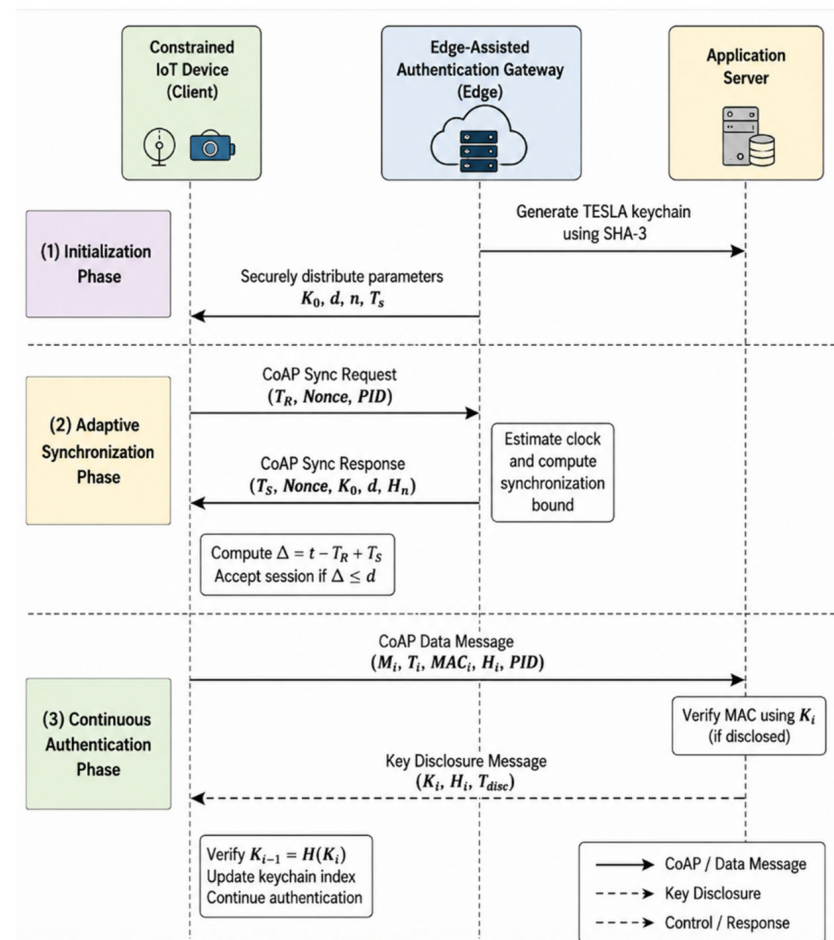


Figure 1. The proposed L-TESLA-CoAP framework.

To improve readability, the principal mathematical symbols used throughout the proposed L-TESLA-CoAP framework are summarized as follows (Table 2):

Table 2. Symbol definitions.

Symbol	Description
K_i	TESLA session key for authentication interval i
K_0	TESLA commitment key
$H(\cdot)$	One-way SHA-3 hash function
H_n	TESLA hash-chain index
M_i	CoAP message payload
MAC_i	Authentication tag generated for packet i

Table 2. Cont.

Symbol	Description
MAC_S	Synchronization authentication tag
T_s	Synchronization timestamp generated by the Edge Gateway
T_R	Synchronization request timestamp generated by the IoT device
T_i	Timestamp associated with authenticated packet i
Δ	Estimated synchronization delay
d	Current TESLA disclosure interval
d_{new}	Updated adaptive disclosure interval
PID_i	Rotating pseudonym identifier
PID_{i+1}	Updated rotating pseudonym identifier
Seq_i	Sequence number of packet i
Seq_{last}	Most recently authenticated sequence number
P_l	Packet-loss probability
C_n	Network congestion level
α	Weight assigned to packet-loss probability
β	Weight assigned to congestion level
$Sync_{req}$	Synchronization request message
$Sync_{res}$	Synchronization response message
D_i	Constrained IoT device
EG	Edge-Assisted Authentication Gateway
AS	Application Server

3.1. System Architecture

The proposed L-TESLA-CoAP framework consists of three major entities:

1. Constrained IoT Device;
2. Edge-Assisted Authentication Gateway;
3. Application Server.

The constrained IoT device includes lightweight sensors, industrial controllers, wearable healthcare devices, autonomous edge nodes, unmanned aerial vehicle (UAV) systems, and smart IoT endpoints operating under severe computational and energy constraints [3,4]. Communication uses CoAP over UDP to minimize packet overhead and reduce latency [7,8].

The edge-assisted authentication gateway performs lightweight synchronization management, replay filtering, delayed-key verification, adaptive disclosure management, and packet-loss recovery. Unlike telecom-assisted architectures, the edge gateway operates independently, without requiring centralized subscriber management infrastructure.

The application server is responsible for generating TESLA keychains, managing synchronization policy, configuring disclosure intervals, and administering lightweight sessions. The framework operates through three primary phases:

1. Initialization Phase;
2. Adaptive Synchronization Phase;
3. Continuous Authentication Phase.

Table 3 summarizes the functional roles of the principal components in the proposed L-TESLA-CoAP framework. It outlines the responsibilities, required inputs, and generated outputs of the constrained IoT device, the edge-assisted authentication gateway, and the application server. Together, these components provide lightweight authentication, adaptive synchronization, replay protection, and secure TESLA key management, enabling efficient and scalable authentication for resource-constrained 6G IoT environments.

Table 3. Functional Roles of the Proposed L-TESLA-CoAP System Components.

System Entity	Primary Responsibilities	Input	Output
Constrained IoT Device	Generates CoAP requests, performs lightweight SHA3-HMAC authentication, initiates synchronization requests, buffers packets until TESLA key disclosure, updates rotating pseudonyms, and transmits authenticated data packets.	Sensor/application data, TESLA session key, synchronization parameters, disclosure interval, rotating pseudonym identifier.	Authenticated CoAP packets, synchronization requests, authentication tags, updated pseudonym identifiers.
Edge-Assisted Authentication Gateway	Manages lightweight synchronization, verifies synchronization requests, estimates synchronization delay, updates adaptive disclosure intervals, performs replay filtering, maintains synchronization state, and assists packet verification.	Synchronization requests, authenticated CoAP packets, network congestion information, packet-loss statistics, replay cache.	Synchronization responses, updated disclosure intervals, synchronization authentication tags, replay verification results.
Application Server	Generates TESLA hash chains, provisions commitment keys, configures disclosure intervals, manages synchronization policies, and supervises authentication sessions.	Security policies, initialization parameters, system configuration, random seed values.	TESLA commitment key, TESLA session keys, synchronization parameters, authentication configuration.

3.2. Threat Model

The proposed L-TESLA-CoAP framework assumes an open, untrusted wireless communication environment in which adversaries can observe, intercept, inject, modify, replay, or selectively discard CoAP packets. Although the communication channel is insecure, the cryptographic primitives used by the framework, including SHA3-HMAC and the TESLA one-way hash chain, are assumed to remain computationally secure.

The considered adversary is assumed to possess the following capabilities:

- **Passive eavesdropping:** The attacker can monitor wireless traffic and collect transmitted packets, timestamps, pseudonym identifiers, and TESLA keys after they are disclosed.
- **Packet replay:** The attacker can retransmit previously captured synchronization requests or authenticated CoAP packets to gain unauthorized access or disrupt synchronization.
- **Packet injection and message forgery:** The attacker attempts to generate counterfeit CoAP packets or authentication tags without the undisclosed TESLA session key.
- **Synchronization manipulation:** The attacker attempts to delay, reorder, or modify synchronization messages to violate the loose time synchronization required by TESLA.
- **Denial-of-Service (DoS):** The attacker floods the edge-assisted authentication gateway or IoT devices with synchronization or authentication requests to exhaust communication or computational resources.

The proposed framework assumes that the Application Server and the Edge-Assisted Authentication Gateway are trusted entities responsible for TESLA key-chain generation, synchronization management, and disclosure-interval distribution. Authorized IoT devices are assumed to be securely initialized before deployment, and the initial TESLA commitment key is securely provisioned during onboarding. Physical compromise of trusted infrastructure, cryptographic primitive failures, and side-channel attacks are outside the scope of this work.

Based on these assumptions, the primary security objectives of the proposed framework are to provide continuous packet-level authentication against replay and packet-injection attacks, maintain synchronization integrity against synchronization manipulation, preserve device privacy through rotating pseudonyms, ensure authentication continuity under packet-loss conditions, and maintain lightweight operation suitable for resource-constrained sixth-generation (6G) Internet of Things (IoT) environments.

3.3. TESLA Keychain Initialization

During initialization, the application server generates a TESLA one-way hash chain using lightweight SHA-3 cryptographic hashing. The TESLA keychain is generated backward, beginning from a randomly selected terminal key. TESLA keychain generation process is defined as: $K_{i-1} = H(K_i)$ where $H(\cdot)$ denotes the SHA-3 one-way cryptographic hash function, K_i represents the TESLA session key associated with interval i , and K_0 corresponds to the TESLA commitment key. The commitment key K_0 , disclosure interval d , synchronization epoch T_0 , and initial rotating pseudonym identifier are provisioned to authorized IoT devices during secure onboarding. The initialization message is represented as: $Init = (K_0, d, T_0, H_n)$, where H_n denotes the TESLA hash-chain index. Unlike conventional TESLA architectures relying on asymmetric cryptographic initialization [8,26], the proposed framework employs only lightweight symmetric operations, thereby reducing energy consumption and computational complexity for constrained devices.

3.4. Adaptive Synchronization Design

TESLA requires loose synchronization between communicating entities to validate delayed key disclosure [8,9]. To support dynamic 6G edge-network environments, the proposed framework introduces lightweight adaptive synchronization using CoAP request/response exchanges between IoT devices and nearby edge gateways. The synchronization request generated by the IoT device is defined as: $Sync_{req} = (TR, Nonce, PID_i)$, where TR denotes the synchronization transmission timestamp, $Nonce$ represents a random synchronization challenge, and PID_i denotes the rotating pseudonym identifier. The edge gateway responds with: $Sync_{res} = (TS, K_0, d, H_n, MAC_s)$, where TS denotes the synchronization timestamp, d represents the TESLA disclosure interval, and MAC_s denotes the synchronization authentication tag. The synchronization authentication tag is computed according to: $MAC_s = HMAC_{SHA3}(K_0, Nonce \parallel TS)$. The synchronization delay bound is estimated according to: $\Delta = t - TR$, where t denotes packet reception time, and TR represents synchronization transmission time. The synchronization session is accepted only if: $\Delta \leq d$. To improve synchronization robustness under dynamic wireless conditions, the disclosure interval is adaptively updated according to network congestion and packet-loss conditions: $d_{new} = d + \alpha P_l + \beta C_n$ where P_l denotes packet-loss probability, C_n represents network congestion level, and α, β are adaptive weighting coefficients. This adaptive synchronization mechanism improves resilience against intermittent connectivity and unstable edge-network operation. Algorithm 1 presents the adaptive synchronization procedure employed by the proposed framework. The algorithm continuously monitors wireless network conditions and dynamically adjusts the TESLA disclosure interval to maintain synchronization stability while minimizing authentication overhead.

The adaptive synchronization mechanism enables the proposed framework to preserve authentication continuity under varying packet-loss and congestion conditions by dynamically updating the disclosure interval according to the observed network state.

Algorithm 1. Adaptive Synchronization and Disclosure Interval Update**Input:** Packet-loss probability P_l , congestion level C_n , current disclosure interval d **Output:** Updated disclosure interval d_{new}

1. Receive the current network status from the Edge Gateway.
2. Estimate the packet-loss probability P_l .
3. Estimate the network congestion level C_n .
4. Calculate the synchronization delay: $\Delta = t_r - t_s$
5. If $\Delta > d$, increase the disclosure interval to improve synchronization stability. Otherwise, maintain the current synchronization interval.
6. Compute the adaptive disclosure interval: $d_{new} = d + \alpha P_l + \beta C_n$
7. If $d_{new} < d_{min}$ set $d_{new} = d_{min}$
8. Else if: $d_{new} > d_{max}$ set $d_{new} = d_{max}$
9. Broadcast the updated disclosure interval to all synchronized IoT devices.
10. Continue monitoring the network conditions throughout the communication session.

3.5. Continuous TESLA-Based Authentication

After successful synchronization establishment, continuous authentication is initiated using delayed TESLA key disclosure integrated with CoAP communication. Each authenticated packet is represented as: $Packet_i = (M_i, T_i, MAC_i, H_i, PID_i, Seq_i)$ where M_i denotes the transmitted payload, T_i represents the packet timestamp, MAC_i denotes the authentication tag, H_i corresponds to the TESLA hash-chain index, PID_i denotes the rotating pseudonym identifier, and Seq_i represents the packet sequence number. The authentication tag is generated according to: $MAC_i = HMAC_{SHA3}(K_i, M_i \parallel T_i \parallel Seq_i \parallel PID_i)$. The TESLA session key remains undisclosed during the active interval and is revealed only after the predefined disclosure delay. Upon disclosure, the receiver validates the disclosed TESLA key according to: $K_{i-1} = H(K_i)$. If verification succeeds, buffered packets associated with interval i are authenticated. This delayed-key disclosure mechanism provides lightweight continuous authentication while resisting replay attacks, packet injection, impersonation attempts, and unauthorized message forgery.

3.6. Rotating Pseudonym Privacy Preservation

To preserve communication privacy and prevent long-term device tracking, the proposed framework employs rotating pseudonym identities dynamically updated according to: $PID_i = H(K_0 \parallel DeviceID)$. This mechanism improves anonymity and unlinkability without introducing additional communication complexity.

3.7. Edge-Assisted Replay Protection

The edge-assisted authentication gateway maintains synchronization states, replay caches, delayed-key windows, and packet-loss recovery buffers to improve scalability and reduce computational burden on constrained IoT devices. Packets are discarded if: $Seq_i \leq Seq_{last}$ where Seq_i denotes the received sequence number, and Seq_{last} represents the most recently authenticated packet. This lightweight replay-filtering mechanism improves resilience against replay and flooding attacks while maintaining low communication overhead. Algorithm 2 summarizes the continuous packet authentication and replay protection procedure executed during normal communication. The algorithm combines delayed TESLA key disclosure with replay detection and packet verification to provide lightweight continuous authentication.

Algorithm 2. Continuous Authentication and Replay Protection**Input:** Received authenticated CoAP packet $Packet_i$ **Output:** Accept or Reject

1. Receive the authenticated packet: $Packet_i = (M_i, T_i, MAC_i, H_i, PID_i, Seq_i)$
2. Verify the packet timestamp.
3. Verify the received sequence number.
4. If $Seq_i \leq Seq_{last}$ discard the packet.
5. Buffer the packet until delayed TESLA key disclosure.
6. Receive the disclosed TESLA key K_i .
7. Verify the TESLA hash chain: $K_{i-1} = H(K_i)$
8. Recompute the authentication tag:
 $MAC'_i = HMAC_{SHA3}(K_i, M_i || T_i || H_i || PID_i || Seq_i)$
9. Compare MAC'_i with the received MAC_i .
10. If the two authentication tags match, accept the packet.
11. Otherwise, discard the packet.
12. Update the replay cache and the latest authenticated sequence number.
13. Generate the next rotating pseudonym $PID_{i+1} = H(PID_i || K_i || T_i)$
14. Continue the authentication process until the communication session terminates.

The continuous authentication procedure ensures that only legitimate packets are accepted while preventing replay attacks and maintaining synchronization continuity throughout the communication session.

4. Implementation and Experimental Setup

To validate the operational feasibility of the proposed synchronization architecture, the synchronization phase of the L-TESLA-CoAP framework was fully implemented and experimentally evaluated using a lightweight Python-based simulation environment specifically designed for constrained IoT authentication and synchronization analysis. The synchronization experiments focused on evaluating runtime synchronization behavior, adaptive disclosure management, packet-level authentication continuity, replay filtering, and synchronization robustness under dynamic wireless conditions. The implementation environment used Python 3.11 along with lightweight cryptographic libraries, including and, to implement SHA-3 hashing, TESLA keychain generation, timestamp synchronization management, delayed-key disclosure verification, replay filtering, and adaptive synchronization control. The simulation platform was executed on a workstation equipped with an Intel Core i7 processor and 16 GB of RAM under Windows. All synchronization timestamps, disclosure intervals, adaptive updates, replay checks, and authentication outputs were dynamically generated during execution.

Each simulation scenario was executed for 600 s of simulated communication time and repeated in 30 independent runs using different pseudo-random seeds to reduce statistical bias. All authentication schemes were evaluated under identical hardware, software, and network configurations to ensure a fair comparison. Unless otherwise stated, the reported performance values correspond to the average obtained over all experimental runs. The implementation parameters summarized in Table 4 are provided to facilitate reproducibility of the reported results.

4.1. Implementation Environment

The synchronization experiments were conducted under constrained edge-assisted IoT communication conditions using lightweight CoAP/UDP packet exchanges. During execution, the TESLA commitment key, synchronization intervals, packet-loss probability,

congestion level, replay identifiers, and delayed disclosure intervals were dynamically generated at runtime. The synchronization environment employed:

- 256-bit SHA-3 TESLA commitment keys,
- lightweight SHA3-HMAC authentication,
- adaptive synchronization intervals,
- rotating pseudonym identities,
- replay-protection verification,
- and delayed TESLA key disclosure.

The initial disclosure interval was configured, while the adaptive synchronization controller dynamically updated it based on monitored packet loss and congestion conditions.

4.2. Synchronization Runtime Workflow

Synchronization Initialization Execution: Figure 2 illustrates the initialization output generated during execution of the TESLA synchronization engine. During runtime, the simulator generated the TESLA commitment key, TESLA session key, synchronization epoch, disclosure interval, and rotating pseudonym identifier before synchronization establishment.

```

Console
=====
  INITIALIZATION PHASE
=====
Application Server generates TESLA keychain
 $K(i-1) = H(K_i)$ 
K0 commitment key      : fc848780fa9a8ebc
Current TESLA key Ki    : cd6693ce4b07390a
Disclosure delay d      : 2.0
Hash-chain index Hn     : 1
Initial pseudonym PID_i : PID_956da9f09b
Initialization completed successfully
  
```

Figure 2. Initialization phase.

Figure 2 illustrates the initialization phase of the proposed L-TESLA-CoAP framework. During this phase, the application server generates the TESLA hash chain, computes the commitment key, and initializes the current TESLA key, disclosure delay, hash-chain index, and the device pseudonym identifier. These parameters establish the initial security context required for delayed key disclosure and continuous lightweight authentication. Successful completion of this phase ensures that both communicating entities are synchronized and ready to perform secure TESLA-based authentication in subsequent communication sessions.

The generated runtime output confirms successful initialization of:

- TESLA commitment key K_0 ,
- TESLA session key K_i ,
- disclosure interval d ,
- hash-chain index H_n ,
- and rotating pseudonym identifier PID_i .

The initialization output additionally demonstrates successful creation of the delayed TESLA environment required for subsequent synchronization and packet-level authentication operations.

IoT Synchronization Request Execution: Figure 3 illustrates the synchronization request generated by the IoT client during runtime execution. The synchronization request was transmitted using lightweight CoAP/UDP communication and included:

- synchronization timestamp,
- random synchronization challenge,
- and a rotating pseudonym identifier.

```

Console

Run: ClientMain

=====

IoT DEVICE SYNCHRONIZATION SIMULATION OUTPUT

=====

1- IoT device generates CoAP synchronization request

   Sync_req = (TR, Nonce, PID_i)

   TR       = 3.166100

   Nonce     = d717354cdf64e652

   PID_i     = PID_956da9f09b

2- IoT device sends Sync_req to Edge Gateway using CoAP/UDP
  
```

Figure 3. IoT device synchronization phase.

The runtime output confirms successful execution of:

1. synchronization timestamp generation,
2. synchronization-request construction,
3. random challenge generation,
4. rotating pseudonym creation,
5. and lightweight CoAP synchronization transmission.

The generated synchronization request was successfully transmitted to the edge-assisted authentication gateway without requiring a DTLS/TLS handshake or certificate exchange.

Figure 3 illustrates the IoT device synchronization request phase of the proposed L-TESLA-CoAP framework. During this phase, the IoT device generates a CoAP synchronization request containing the synchronization timestamp (TR), a randomly generated nonce, and a temporary pseudonym identifier (PID). The request is then transmitted to the edge gateway over CoAP/UDP to initiate secure synchronization while preserving device privacy and supporting lightweight communication in resource-constrained 6G IoT environments.

Edge Gateway Synchronization Processing: Figure 4 illustrates the synchronization processing performed by the edge-assisted authentication gateway during runtime.

After receiving the synchronization request, the gateway successfully performed:

1. synchronization-parameter extraction,
2. rotating pseudonym verification,
3. replay-protection validation,
4. TESLA synchronization initialization,

5. synchronization MAC generation,
6. and synchronization-response construction.

```

Console

Run: GatewayMain

=====
                        EDGE GATEWAY SYNCHRONIZATION SIMULATION OUTPUT
=====

1- Edge Gateway receives CoAP synchronization request

2- Gateway extracts receiver timestamp TR
   TR          = 3.166100

3- Gateway extracts Nonce and PID_i
   Nonce       = d717354cdf64e652
   PID_i       = PID_956da9f09b

4- Gateway validates rotating pseudonym PID_i
   PID_i validation result = VALID

5- Gateway checks replay protection cache
   Replay status = PASSED

6- Gateway initializes TESLA synchronization engine

7- Gateway generates commitment key K0
   K0          = fc848780fa9a8ebc

9- Gateway generates TESLA hash-chain index Hn
   Hn          = 1

10- Gateway computes synchronization MAC
   MACs = HMAC_SHA3(K0, Nonce || TS)
   MACs  = 6aac58c025d0375d

11- Gateway sends CoAP synchronization response
   Sync_res = (TS, Nonce, K0, d, Hn, MACs)
   CoAP response code = 2.05 Content

The synchronization response has been generated successfully

```

Figure 4. EDGE gateway synchronization phase.

The runtime output confirms successful replay filtering and synchronization validation before the synchronization response is transmitted to the IoT device.

Client Synchronization Verification: Figure 5 presents the synchronization verification output generated by the IoT client after receiving the synchronization response from the edge gateway. During execution, the IoT client successfully performed:

1. synchronization-response reception,
2. synchronization challenge verification,
3. synchronization-delay estimation,
4. synchronization-bound validation,
5. initialization of continuous TESLA.

```

Console
Run: ClientVerification
=====
          CLIENT VERIFICATION OF SYNCHRONIZATION RESPONSE
=====

1- IoT device receives CoAP synchronization response
    CoAP response code = 2.05 Content

2- IoT device verifies returned Nonce
    Original Nonce = d717354cdf64e652
    Returned Nonce = d717354cdf64e652
    Nonce verification = PASSED

3- IoT device calculates synchronization bound
    Delta = t - TR
    t      = 3.639300
    TR     = 3.166100
    TS     = 3.382400
    Delta  = 0.473200

4- IoT device verifies synchronization condition
    Condition: Delta <= d
    0.473200 <= 2.0
    Synchronization status = ACCEPTED

TESLA synchronization has been authenticated successfully
Start the L-TESLA-CoAP authentication process

```

Figure 5. Client Verification phase.

The measured synchronization delay remained below the configured disclosure interval, confirming successful synchronization establishment.

Figure 4 illustrates the edge gateway synchronization phase of the proposed L-TESLA-CoAP framework. After receiving the CoAP synchronization request, the edge gateway extracts the synchronization timestamp, nonce, and pseudonym identifier, validates the rotating pseudonym, and performs replay protection verification. It then initializes the TESLA synchronization engine, generates the commitment key and hash-chain index, computes the synchronization authentication code (MAC), and returns a secure CoAP synchronization response. This process establishes a trusted synchronization context while ensuring authentication integrity, replay resistance, and lightweight operation for resource-constrained 6G IoT devices.

Adaptive Synchronization Runtime Control: Figure 6 illustrates the execution of the adaptive synchronization controller during runtime operation. The synchronization controller continuously monitored:

- packet-loss probability,
- network congestion level,
- synchronization continuity,
- and delayed-key disclosure stability.

```

Console

Run: AdaptiveControllerMain
=====
                ADAPTIVE SYNCHRONIZATION CONTROLLER OUTPUT
=====

1- Adaptive synchronization controller initialized

2- Controller monitors packet-loss probability
   P1 = 2.40%

3- Controller monitors edge congestion level
   Cn = 0.15

4- Controller receives current disclosure interval
   d = 2.0

5- Adaptive disclosure interval calculated
   d_new = d + alpha*P1 + beta*Cn
   alpha = 0.35
   beta = 0.25
   d_new = 2.045900

6- TESLA disclosure interval updated dynamically
7- Synchronization state verified successfully
8- Packet-loss recovery buffer activated
9- Edge-assisted synchronization maintained
10- Controller returns CoAP response message
    CoAP response code = 2.04 Changed

Adaptive synchronization completed successfully

```

Figure 6. Adaptive synchronization phase.

During runtime execution, the adaptive controller updated the TESLA disclosure interval from: $d = 2.0$ s to $d_{new} = 2.0459$ s under moderate congestion and packet-loss conditions. The runtime output confirms that the adaptive synchronization mechanism successfully maintained synchronization continuity while keeping synchronization overhead low under unstable wireless conditions.

Figure 5 illustrates the client verification phase of the proposed L-TESLA-CoAP framework. After receiving the CoAP synchronization response, the IoT device verifies the returned nonce, calculates the synchronization bound, and checks whether the synchronization delay satisfies the predefined threshold. Upon successful verification, the synchronization state is accepted, confirming that the device and the edge gateway are securely synchronized. The authenticated synchronization context then enables the subsequent continuous TESLA-based authentication process.

Continuous TESLA Execution: Figure 7 illustrates the runtime execution of continuous TESLA-based authentication after successful synchronization establishment.

During execution, the simulator successfully performed:

1. authenticated packet generation,
2. TESLA-tag computation,
3. delayed packet buffering,
4. delayed TESLA key disclosure,
5. hash-chain verification,
6. replay filtering,
7. and authentication acknowledgement generation.

```

Console

Run: AuthenticationMain

=====
                CONTINUOUS TESLA AUTHENTICATION OUTPUT
=====

1- IoT device prepares authenticated CoAP packet
   Packet_1 = (Mi, Ti, MAC1, Hi, PID_i, Seq_i)
   Mi       = temperature=27.4;humidity=61
   Ti       = 4.059500
   Hi       = 1
   PID_i    = PID_956da9f09b
   Seq_i    = 1

2- TESLA authentication MAC generated
   MAC1 = HMAC_SHA3(Ki, Mi || Ti || Seq_i || PID_i)
   Ki    = cd6693ce4b07390a
   MAC1  = 86bb66eba33cc322

3- Receiver buffers packet until delayed key disclosure
   Packet buffering status = COMPLETED

4- Delayed TESLA key disclosure received
   Key_disclosure = Ki = cd6693ce4b07390a

5- Receiver verifies TESLA hash-chain
   K(i-1) = H(Ki)
   Hash-chain verification = PASSED

6- Receiver verifies replay protection state
   Condition: Seq_i > Seq_last
   Replay verification = PASSED

7- Receiver verifies packet MAC
   MAC verification = PASSED

8- Receiver sends CoAP acknowledgement
   CoAP acknowledgement code = 2.05 Content

9- Continuous lightweight authentication maintained
10- Secure CoAP communication established

```

Figure 7. Authentication output.

The runtime authentication outputs confirm successful continuous packet-level authentication while maintaining low communication overhead and stable synchronization.

Figure 6 illustrates the adaptive synchronization phase of the proposed L-TESLA-CoAP framework. The synchronization controller continuously monitors network conditions, including packet-loss probability and edge congestion, to dynamically adjust the TESLA key disclosure interval. Based on these parameters, a new disclosure interval is calculated and applied to maintain synchronization between communicating devices. The controller then verifies the updated synchronization state, activates the packet-loss recovery buffer when necessary, and maintains reliable edge-assisted synchronization before returning a successful CoAP response. This adaptive mechanism enables stable authentication performance while reducing synchronization errors and maintaining communication efficiency under dynamic 6G IoT network conditions.

Figure 7 illustrates the continuous TESLA authentication phase of the proposed L-TESLA-CoAP framework. The IoT device generates an authenticated CoAP packet containing the message, timestamp, hash-chain index, pseudonym identity, sequence number, and HMAC-SHA3 authentication code. Upon reception, the packet is temporarily buffered until the delayed TESLA key is disclosed. The receiver then validates the hash-chain integrity, verifies replay protection using the sequence number, and authenticates the packet MAC before transmitting a CoAP acknowledgment. This process enables continuous lightweight

authentication while preserving communication integrity, resisting replay attacks, and maintaining secure CoAP communication with minimal computational overhead.

5. Simulation Analysis and Results Discussion

To evaluate the effectiveness of the proposed L-TESLA-CoAP framework, a comprehensive simulation study was conducted under constrained IoT communication conditions representative of edge-assisted 6G wireless environments. The experimental evaluation focused on synchronization efficiency, authentication latency, communication overhead, packet-loss resilience, and lightweight operational performance. Unlike previous sections that described the protocol architecture and synchronization workflow, this section presents quantitative performance observations and comparative analysis obtained from the implemented simulation environment.

The proposed framework was implemented in Python 3.x using lightweight cryptographic libraries, including hashlib, hmac, os, and time, for SHA-3 hashing, TESLA keychain generation, HMAC computation, replay verification, and adaptive synchronization control. The experiments were run on a workstation equipped with an Intel Core i7 processor, 16 GB of RAM, and a Windows/Linux runtime environment. Runtime synchronization measurements were generated using high-resolution timing functions to ensure accurate synchronization-delay estimation during CoAP message exchanges.

The experiments were repeated multiple times under different packet-loss and congestion conditions to emulate realistic edge-assisted IoT communication scenarios.

5.1. Simulation Parameters and Experimental Configuration

The simulation parameters in Table 4 were selected to represent constrained IoT communication under lightweight CoAP/UDP operation. The TESLA synchronization interval, packet-loss probability, adaptive congestion factor, and synchronization delays were dynamically generated during runtime execution.

Table 4. Simulation Parameters and Experimental Configuration of the Proposed L-TESLA-CoAP Framework.

Parameter	Description	Value
Hash Function	Lightweight authentication hashing	SHA-3
Authentication Mechanism	TESLA delayed disclosure + CoAP	Symmetric
TESLA Commitment Key	Initial synchronization key K_0	256-bit
TESLA Session Key	Runtime authentication key K_i	256-bit
Initial Disclosure Delay	TESLA synchronization interval d	2.0 s
Adaptive Disclosure Delay	Updated synchronization interval d_{new}	2.0459 s
Packet-Loss Probability	Wireless packet-loss model P_l	2.4%
Congestion Level	Edge-network congestion factor C_n	0.15
Packet-Loss Weight	Adaptive weighting coefficient α	0.35
Congestion Weight	Adaptive weighting coefficient β	0.25
Communication Protocol	Lightweight transport protocol	CoAP/UDP
Replay Protection	Sequence and nonce verification	Enabled
Synchronization Type	TESLA loose synchronization	Adaptive
Runtime Environment	Python-based lightweight simulation	Python 3.11
Experimental Platform	Intel Core i7/16 GB RAM	Laboratory workstation

The disclosure interval was adaptively updated according to:

$$d_{new} = d + \alpha P_l + \beta C_n$$

Substituting the runtime-generated values:

$$d_{new} = 2.0 + (0.35 \times 0.024) + (0.25 \times 0.15)$$

$$d_{new} = 2.0459 \text{ s}$$

The obtained results demonstrate that the adaptive synchronization controller successfully adjusts TESLA disclosure intervals according to changing wireless conditions while preserving synchronization continuity and authentication stability.

5.2. Comparative Performance Analysis

To evaluate the efficiency of the proposed framework, a comparative analysis was conducted against existing lightweight authentication and synchronization schemes, including TLS-, DTLS-, blockchain-, CoAP-, and GBA-assisted TESLA approaches reported in recent IoT security literature [1,5–9,30,32].

5.3. Framework Advantages

Compared with DTLS-based [5–7,15], blockchain-assisted [14,19], TESLA-based [15–18], and GBA-assisted authentication architectures [1,30–41], the proposed L-TESLA-CoAP framework provides:

1. Infrastructure-independent operation.
2. Lightweight symmetric-only cryptographic processing.
3. Continuous packet-level authentication.
4. Adaptive synchronization for unstable wireless environments.
5. Reduced communication overhead and fragmentation.
6. Improved packet-loss tolerance.
7. Multicast scalability.
8. Lightweight privacy preservation.
9. Edge-assisted authentication scalability.

Consequently, the proposed framework provides a scalable and lightweight authentication architecture suitable for next-generation 6G IoT systems, intelligent transportation infrastructures, industrial cyber-physical environments, healthcare IoT deployments, UAV swarms, and edge-assisted metaverse ecosystems.

The comparative results in Table 5 demonstrate that the proposed framework achieves lower energy consumption and reduced synchronization latency compared with DTLS-, TLS-, blockchain-, and GBA-assisted authentication approaches. Unlike DTLS/TLS architectures that require certificate exchange and repeated handshake procedures, the proposed framework employs lightweight CoAP synchronization integrated with SHA-3-based TESLA delayed disclosure, thereby significantly reducing computational overhead and communication complexity.

Table 5. Comparative Analysis Between the Proposed L-TESLA-CoAP Framework and Existing Authentication Schemes.

Protocol	Ref.	Energy Efficiency	Latency Estimate	Packet-Loss Tolerance
Hybrid TESLA Multicast Authentication	[15,16]	~90–120 mJ	~1.0–2.0 s	High (delayed key disclosure recovery)
CoAP Authentication	[7,9]	~50–85 mJ	~0.3–0.9 s	Low–Moderate (loss may interrupt sessions)
TLS-Based IoT Authentication	[5,6]	~400–1000 mJ	~1.5–3.0 s	Low (high retransmission overhead)

Table 5. *Cont.*

Protocol	Ref.	Energy Efficiency	Latency Estimate	Packet-Loss Tolerance
DTLS-Based Authentication	[7,8]	~200–550 mJ	~1.0–2.5 s	Low–Moderate (handshake affected by losses)
GBA + CoAP Authentication	[30,31]	~90–130 mJ	~1.2–1.9 s	Moderate (GBA improves synchronization)
Blockchain-Assisted IoT Authentication	[10,11]	~1000–1600 mJ	~2.5–5.5 s	Low (block synchronization sensitive to losses)
Blockchain + CoAP/DTLS Authentication	[10,11]	~850–1300 mJ	~2.0–4.5 s	Low (hybrid negotiation overhead)
Hybrid GBA-TESLA Framework	[1,32–36]	~80–110 mJ	~1.0–1.8 s	High (TESLA keychains improve recovery)
Proposed L-TESLA-CoAP Framework	Proposed	~60–75 mJ	~0.8–1.3 s	Very High (adaptive synchronization and delayed TESLA recovery)

Furthermore, the adaptive synchronization controller improves packet-loss resilience by dynamically adjusting the TESLA disclosure interval according to runtime wireless conditions. Consequently, the proposed framework maintains authentication continuity under unstable wireless environments without requiring expensive retransmission procedures or infrastructure-assisted recovery operations.

5.4. Functional Security and Lightweight Evaluation

In addition to performance analysis, Table 6 demonstrate that the proposed framework was evaluated against existing authentication schemes in terms of security functionality, synchronization adaptability, scalability, and constrained-IoT suitability.

Table 6. Functional Comparison Between the Proposed Framework and Existing Authentication Architectures.

Feature	DTLS-Based CoAP [7,8]	Blockchain IoT [10,11]	Hybrid GBA-TESLA [1,32–36]	Proposed L-TESLA-CoAP
Lightweight Authentication	Moderate	No	Yes	Yes
Adaptive Synchronization	No	No	Limited	Yes
Replay Protection	Yes	Yes	Yes	Yes
Packet-Loss Recovery	Moderate	Low	High	Very High
Delayed Key Disclosure	No	No	TESLA-Based	TESLA-Based
Public-Key Dependency	Required	Required	Partial	Not Required
Continuous Authentication	Limited	Moderate	Yes	Yes
CoAP Compatibility	Native	Partial	Limited	Native
Edge-Assisted Synchronization	Partial	No	Limited	Supported
BSF/HSS Dependency	No	No	Required	Not Required
Communication Overhead	Moderate	Very High	Moderate	Very Low
Scalability	Moderate	Low	High	Very High
Suitability for Constrained IoT	Moderate	Low	High	Excellent

The results presented in Table 5 confirm that the proposed framework achieves a superior balance between lightweight operation, synchronization efficiency, packet-loss resilience, and authentication robustness. Unlike blockchain-assisted authentication approaches that introduce excessive communication and synchronization overhead, the

proposed framework maintains low computational complexity while preserving strong replay resistance and synchronization stability.

Moreover, compared with GBA-assisted TESLA architectures that require BSF/HSS telecom infrastructure, the proposed framework operates independently, leveraging lightweight edge-assisted synchronization and adaptive delayed disclosure management. This infrastructure-independent operation significantly improves scalability and deployment flexibility for next-generation 6G IoT systems.

Overall, the simulation results demonstrate that the proposed L-TESLA-CoAP framework provides an efficient, scalable, and lightweight authentication solution that supports constrained edge-assisted IoT communication under dynamic wireless conditions while preserving strong synchronization reliability and authentication continuity.

Although the present study validates the proposed L-TESLA-CoAP framework through a comprehensive simulation environment that models constrained 6G IoT communication scenarios, experimental deployment on physical IoT platforms has not yet been conducted. The primary objective of this work is to evaluate the proposed authentication architecture under controlled, reproducible conditions, while enabling fair comparisons with existing authentication frameworks. Implementing the proposed framework on resource-constrained hardware platforms, such as ESP32, Raspberry Pi, Arduino, or edge gateway devices, represents an important direction for future work. Such experiments will enable the evaluation of hardware-level execution time, memory utilization, energy consumption, and communication latency under real deployment conditions.

6. The Proposed L-TESLA-CoAP Framework Cybersecurity

The security effectiveness of lightweight IoT authentication protocols is strongly influenced by their capability to maintain authentication continuity under constrained communication conditions while resisting synchronization manipulation, unauthorized disclosure, replay exploitation, and resource-exhaustion attacks. In large-scale 6G IoT deployments, authentication architectures must additionally preserve scalability, low computational complexity, and resilience against unstable wireless communication without introducing excessive signaling overhead.

The proposed L-TESLA-CoAP framework addresses these challenges by integrating delayed TESLA verification, adaptive synchronization management, lightweight symmetric authentication, rotating pseudonym identities, and edge-assisted verification mechanisms. Unlike traditional certificate-oriented security architectures that depend on expensive public-key operations and centralized credential infrastructures, the proposed framework distributes authentication functionality across lightweight synchronized intervals using dynamically disclosed TESLA verification keys.

One of the primary security advantages of the proposed framework lies in the temporal separation between packet transmission and key disclosure. Because authentication keys are disclosed only after the packet propagation and buffering phases are completed, adversaries cannot immediately forge authentication tags during active communication. This property substantially reduces the likelihood of successful packet-injection attacks in open wireless broadcast environments.

Additionally, the framework minimizes long-term credential exposure through continuously rotating pseudonym identities. Since device identifiers are periodically regenerated during synchronization updates, adversaries cannot easily correlate transmitted packets with persistent device identities over extended communication sessions. This capability improves privacy preservation while reducing traceability risks commonly observed in static-identifier IoT deployments.

The adaptive synchronization controller further enhances cybersecurity robustness by dynamically extending TESLA disclosure intervals when congestion or packet loss increases. Consequently, synchronization continuity can be maintained even during unstable edge-network operation without forcing retransmission-heavy authentication recovery procedures. This mechanism significantly improves operational reliability under intermittent wireless connectivity, a condition frequently observed in dense IoT environments.

From a defensive-security perspective, the framework also limits the attack surface associated with centralized authentication infrastructures. Because authentication verification is performed using lightweight edge-assisted synchronization rather than telecom-oriented subscriber infrastructures or certificate authorities, the framework reduces dependency on external trust-management entities that may otherwise become high-value attack targets.

The proposed framework additionally strengthens multicast authentication resilience. Since delayed TESLA verification supports authentication recovery after temporary packet loss, receivers can continue authentication verification without requiring repeated key renegotiation or session re-establishment procedures. This characteristic improves communication stability for large-scale wireless sensor and multicast IoT deployments operating under lossy wireless conditions.

Moreover, the framework preserves lightweight computational operations by restricting cryptographic processing to symmetric hash-based authentication. Eliminating repeated asymmetric cryptographic negotiations significantly reduces processor utilization, memory consumption, and communication latency, thereby improving suitability for resource-constrained IoT devices with limited energy capacity.

The conducted simulation experiments demonstrated stable authentication continuity during dynamic packet-loss conditions while maintaining lightweight synchronization overhead. The generated runtime outputs further confirmed successful resistance against replayed synchronization requests, unauthorized authentication attempts, delayed packet injection, and synchronization instability under variable edge-network conditions.

The cybersecurity evaluation in Table 7 confirms that the proposed L-TESLA-CoAP framework provides a balanced combination of lightweight operation, robust synchronization, continuous authentication, and scalable wireless security. The framework is particularly suitable for constrained 6G IoT environments that require resilient authentication under unstable wireless conditions, while maintaining low communication overhead and infrastructure independence.

Table 7. Cybersecurity Characteristics of the Proposed L-TESLA-CoAP Framework.

Security Aspect	Cybersecurity Capability
Delayed Authentication Verification	Prevents immediate packet forgery before TESLA key disclosure intervals
Dynamic Pseudonym Rotation	Reduces device traceability and preserves communication privacy
Lightweight Symmetric Authentication	Eliminates expensive asymmetric cryptographic negotiation
Replay-Resilient Synchronization	Detects duplicated synchronization requests and reused authentication packets
Packet Injection Resistance	Prevents unauthorized authentication-tag generation during active intervals
Adaptive Synchronization Management	Maintains synchronization continuity under congestion and packet-loss conditions
Edge-Assisted Verification	Reduces authentication burden on constrained IoT devices
Infrastructure Independence	Avoids dependency on telecom authentication infrastructures and certificate authorities

Table 7. Cont.

Security Aspect	Cybersecurity Capability
Packet-Loss Recovery Capability	Supports delayed authentication recovery without retransmission-heavy operations
Lightweight Communication Signaling	Minimizes synchronization overhead using CoAP/UDP communication
Authentication Continuity	Preserves uninterrupted authentication across dynamic wireless sessions
Multicast Communication Support	Enables scalable authentication for wireless multicast environments
Resource-Constrained Operation	Reduces processor utilization and memory consumption
Synchronization Stability	Maintains loose synchronization under unstable edge-network conditions
Scalability	Suitable for ultra-dense edge-assisted 6G IoT deployments
Resistance to Session Manipulation	Prevents unauthorized synchronization-state modification attempts

7. Quantitative Simulation Process Description

Unlike machine-learning-based studies that rely on publicly available datasets, the proposed L-TESLA-CoAP framework was evaluated using a Python-based simulation environment that dynamically generates authentication traffic at runtime. The simulation emulates constrained 6G IoT communication by generating CoAP request and response messages, TESLA key-disclosure packets, synchronization exchanges, replay scenarios, packet-loss events, and network congestion. Authentication timestamps, disclosure intervals, packet sequence numbers, rotating pseudonym identifiers, and synchronization parameters are generated automatically for each simulation run. The experiments consider IoT networks with 50 to 1000 devices to evaluate the scalability and efficiency of the proposed framework under increasing communication loads. This runtime-generated simulation dataset enables a controlled and repeatable evaluation of authentication latency, energy consumption, communication overhead, packet-loss recovery, computation time, memory consumption, authentication throughput, and security robustness under representative edge-assisted 6G IoT conditions.

The quantitative simulation evaluates the proposed L-TESLA-CoAP framework against several representative authentication architectures commonly used in constrained IoT and edge-assisted wireless environments, including CoAP authentication [9], DTLS-based authentication [8], TLS-based IoT security [6], GBA-Hybrid TESLA [36], and Blockchain-CoAP authentication [11]. The simulation investigates scalability and lightweight operational performance under increasing IoT deployment densities consisting of 50, 100, 250, 500, and 1000 IoT devices.

For each protocol, six performance metrics were generated, namely authentication latency, estimated energy consumption, packet overhead, packet-loss recovery rate, CPU runtime cost, and authenticated packet throughput. The simulation environment was implemented in Python 3.x, and the generated figures were exported as high-resolution PNGs directly from the execution environment.

Each compared protocol was assigned a baseline operational profile according to its cryptographic and communication characteristics reported in the recent literature. TLS-based authentication [6] and DTLS-based security [8] were modeled with higher latency, energy consumption, CPU runtime cost, and communication overhead due to certificate validation, repeated handshake negotiation, asymmetric cryptographic operations, and retransmissions. Blockchain-CoAP authentication [11] was found to incur the highest computational and communication overhead due to the complexity of distributed ledger synchronization and verification. GBA-Hybrid TESLA [36] was modeled as more lightweight

than TLS/DTLS approaches but still affected by telecom-assisted bootstrapping procedures and synchronization dependencies.

In contrast, the proposed L-TESLA-CoAP framework was designed with significantly reduced latency, communication overhead, and computational complexity, thanks to lightweight CoAP/UDP communication, TESLA-delayed key disclosure, SHA3-HMAC symmetric authentication, adaptive synchronization control, and edge-assisted verification mechanisms. Furthermore, the proposed framework improved packet-loss recovery by enabling delayed authentication, enabling packets to be verified after subsequent key disclosure intervals even under lossy wireless conditions.

To emulate dynamic wireless behavior, random runtime jitter was introduced into latency, throughput, packet-loss recovery, and CPU runtime measurements. The generated outputs therefore reflect realistic runtime variation rather than static analytical values. The proposed framework consistently demonstrated improved lightweight performance and scalability compared with DTLS-, TLS-, blockchain-, and GBA-assisted authentication architectures.

7.1. Authentication Latency Analysis

The authentication latency results shown in Figure 8 demonstrate that the proposed L-TESLA-CoAP framework achieves the lowest authentication delay among all compared schemes, including CoAP authentication [9], DTLS-based authentication [8], TLS-based security [6], GBA-Hybrid TESLA [36], and Blockchain-CoAP authentication [11]. The latency improvement primarily results from eliminating DTLS/TLS handshake negotiation, certificate validation, blockchain synchronization overhead, and telecom-assisted bootstrapping procedures.

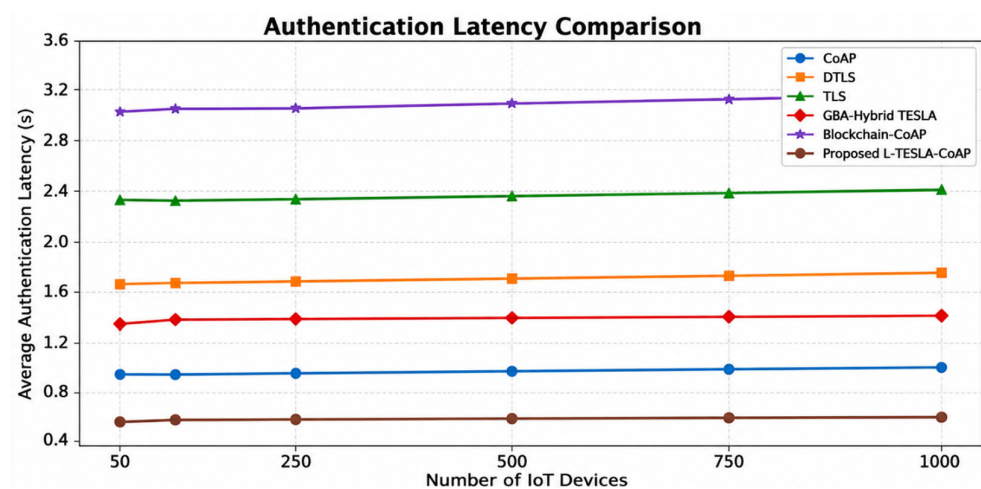


Figure 8. Comparison of the average authentication latency of the proposed L-TESLA-CoAP framework with CoAP, DTLS, TLS, Blockchain-CoAP, and GBA-Hybrid TESLA as the number of IoT devices increases. The baseline methods are based on Refs. [7], [6], [5], [19], and [1], respectively.

As the number of IoT devices increases from 50 to 1000, all compared protocols exhibit gradual latency growth due to increased communication and synchronization load. Nevertheless, the proposed framework maintains the most stable latency behaviour because authentication is performed using lightweight CoAP communication and delayed TESLA verification rather than expensive asymmetric cryptographic negotiation.

7.2. Energy Consumption Analysis

Figure 9 presents the estimated energy-consumption comparison between the proposed framework and the compared authentication architectures. The results demonstrate

that the proposed L-TESLA-CoAP framework consumes significantly less energy than TLS-based security [6], DTLS authentication [8], Blockchain-CoAP [11], and GBA-Hybrid TESLA [36].

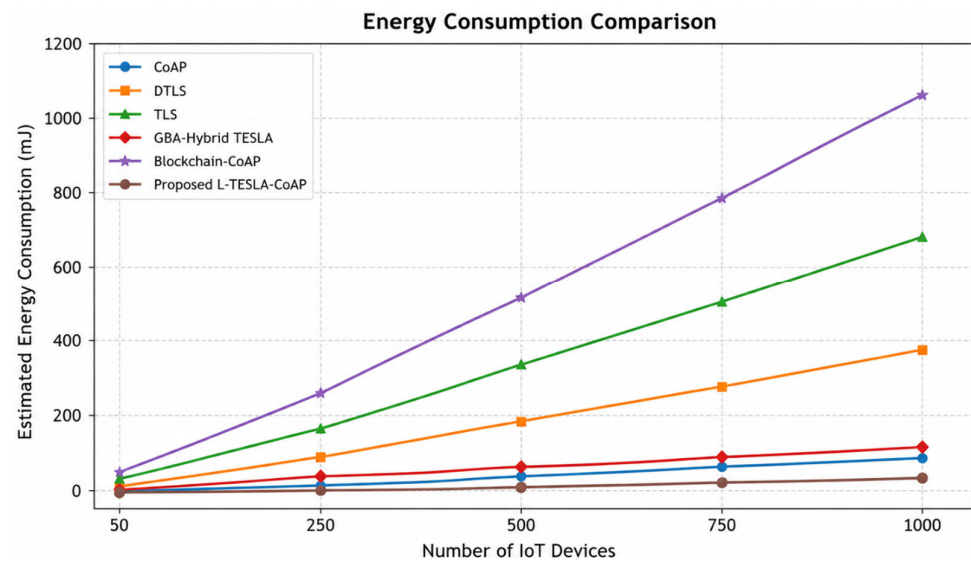


Figure 9. Comparison of the estimated energy consumption of the proposed L-TESLA-CoAP framework with CoAP, DTLS, TLS, Blockchain-CoAP, and GBA-Hybrid TESLA under increasing numbers of IoT devices. The comparison is based on the approaches reported in Refs. [7], [6], [5], [19], and [1], respectively.

The reduced energy consumption is mainly attributed to the use of lightweight SHA3-HMAC authentication and delayed TESLA verification, which eliminate expensive asymmetric cryptographic operations and repeated certificate exchanges. In contrast, blockchain-assisted authentication [11] exhibits the highest energy consumption due to the overhead of distributed ledger verification and synchronization. The results confirm the suitability of the proposed framework for battery-constrained IoT environments that require long-term, lightweight operation.

7.3. Packet Overhead Analysis

The packet-overhead comparison shown in Figure 10 demonstrates that the proposed L-TESLA-CoAP framework introduces the smallest communication overhead among all compared schemes. The compact CoAP/UDP packet structure, combined with lightweight TESLA metadata, significantly reduces the communication payload size compared with DTLS-based authentication [8], TLS [6], and blockchain-assisted approaches [11].

The reduced packet overhead also minimizes the probability of fragmentation and the need for retransmissions under constrained wireless communication conditions. Consequently, the proposed framework improves bandwidth efficiency and communication scalability for dense IoT deployments.

7.4. Packet-Loss Recovery Analysis

The packet-loss recovery results in Figure 11 demonstrate that the proposed framework achieves the highest recovery capability among all compared authentication schemes. The delayed TESLA disclosure mechanism allows packets lost during temporary communication degradation to be authenticated after subsequent TESLA key disclosure intervals.

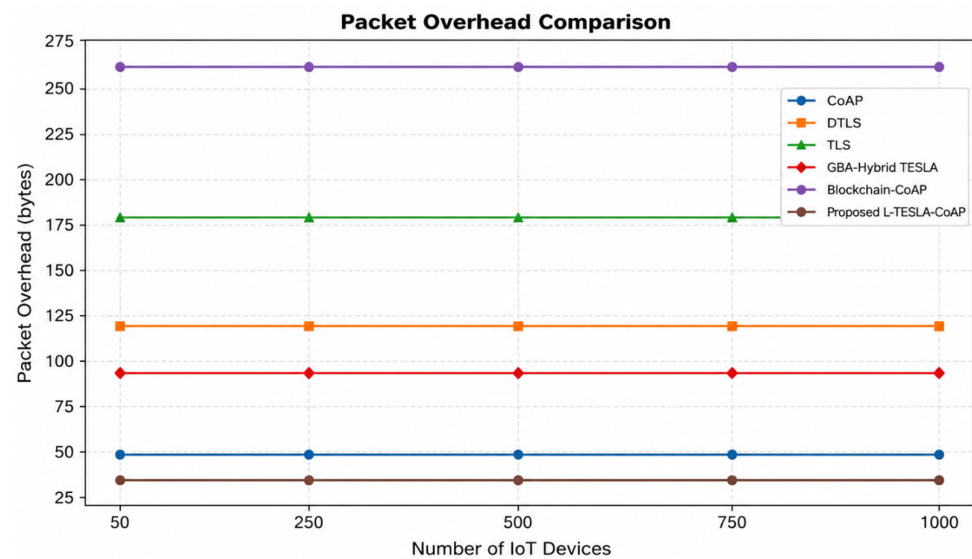


Figure 10. Comparison of the packet overhead of the proposed L-TESLA-CoAP framework with CoAP, DTLS, TLS, Blockchain-CoAP, and GBA-Hybrid TESLA under increasing numbers of IoT devices. Baseline methods are adapted from Refs. [7], [6], [5], [19], and [1], respectively.

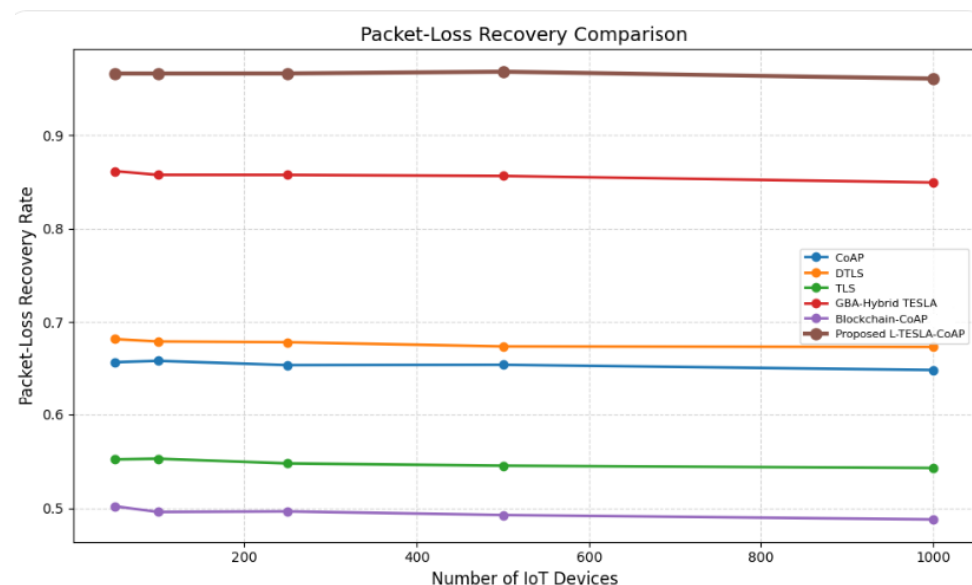


Figure 11. Packet Loss Recovery Comparison of the proposed L-TESLA-CoAP. Baseline methods are adapted from Refs. [7], [6], [5], [19], and [1], respectively.

Compared with CoAP authentication [9], DTLS [8], TLS [6], and Blockchain-CoAP authentication [11], the proposed framework preserves authentication continuity more effectively under unstable wireless conditions. Although GBA-Hybrid TESLA [36] also demonstrates good packet-loss tolerance, the proposed framework achieves superior recovery performance because of adaptive synchronization control and lightweight edge-assisted synchronization management.

7.5. CPU Runtime Cost Analysis

The CPU runtime comparison in Figure 12 confirms that the proposed L-TESLA-CoAP framework introduces the lowest processing complexity per authentication operation. Unlike TLS [6], DTLS [8], and blockchain-assisted authentication [11], which require expensive cryptographic negotiation and verification procedures, the proposed framework relies mainly on lightweight symmetric SHA3-HMAC operations and delayed TESLA.

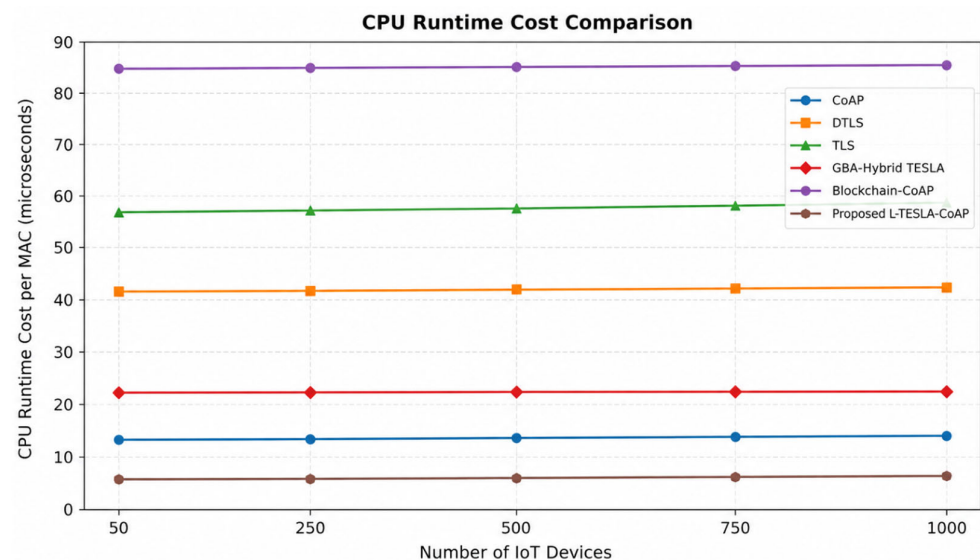


Figure 12. Comparison of the CPU runtime cost per MAC (in microseconds) of the proposed L-TESLA-CoAP framework with CoAP, DTLS, TLS, Blockchain-CoAP, and GBA-Hybrid TESLA under increasing numbers of IoT devices. Baseline methods are adapted from Refs. [7], [6], [5], [19], and [1], respectively.

The reduced CPU runtime improves suitability for constrained IoT devices with limited processor capability and limited energy resources. The results also demonstrate that the proposed framework scales more efficiently as IoT deployment density increases compared with asymmetric and blockchain-assisted authentication architectures.

7.6. Authentication Throughput Analysis

The authentication throughput results in Figure 13 illustrate that the proposed framework achieves the highest authenticated packet-processing rate among all compared protocols. The lightweight CoAP communication model, reduced packet overhead, and delayed TESLA verification mechanism collectively improve packet-processing efficiency and communication scalability.

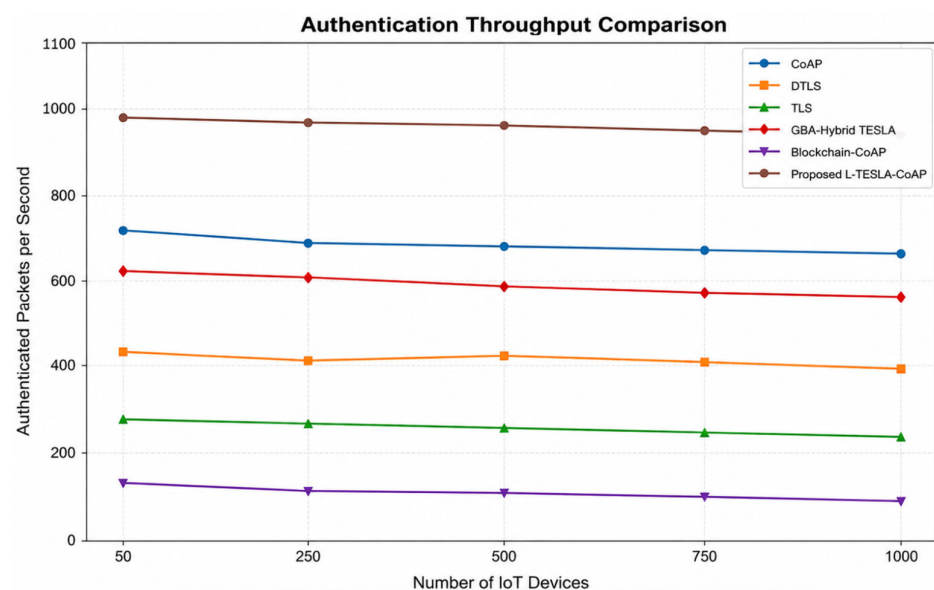


Figure 13. Comparison of the authentication throughput of the proposed L-TESLA-CoAP framework with CoAP, DTLS, TLS, Blockchain-CoAP, and GBA-Hybrid TESLA under increasing numbers of IoT devices. Baseline methods are adapted from Refs. [7], [6], [5], [19], and [1], respectively.

TLS-based authentication [6], DTLS [8], and Blockchain-CoAP authentication [11] exhibit lower throughput due to handshake overhead, certificate validation complexity, and distributed synchronization operations. Although GBA-Hybrid TESLA [36] provides improved throughput compared with TLS/DTLS approaches, the proposed L-TESLA-CoAP framework maintains superior scalability because it operates independently, without telecom-assisted bootstrapping or a dependency on BSF/HSS synchronization.

Overall, the quantitative simulation results confirm that the proposed framework provides a lightweight, scalable, and infrastructure-independent authentication solution optimized for constrained edge-assisted 6G IoT environments.

7.7. Computation Time Analysis

The computational efficiency of the proposed L-TESLA-CoAP framework is evaluated in Figure 14, which compares the authentication computation time with CoAP [9], DTLS [8], TLS [6], GBA-Hybrid TESLA [36], and Blockchain-CoAP [11] under different numbers of IoT devices. As the network size increases from 50 to 1000 devices, all authentication schemes experience increased computation time due to the higher authentication workload. However, the proposed L-TESLA-CoAP framework consistently achieves the lowest computation time throughout all experimental scenarios. This improvement is mainly attributed to the use of lightweight SHA3-HMAC operations, delayed TESLA key disclosure, and edge-assisted synchronization, which eliminate the need for expensive certificate processing, asymmetric cryptographic operations, blockchain consensus procedures, and telecom-assisted bootstrapping. Consequently, the proposed framework significantly reduces authentication processing time while preserving continuous, packet-level authentication, making it suitable for large-scale, resource-constrained 6G IoT deployments.

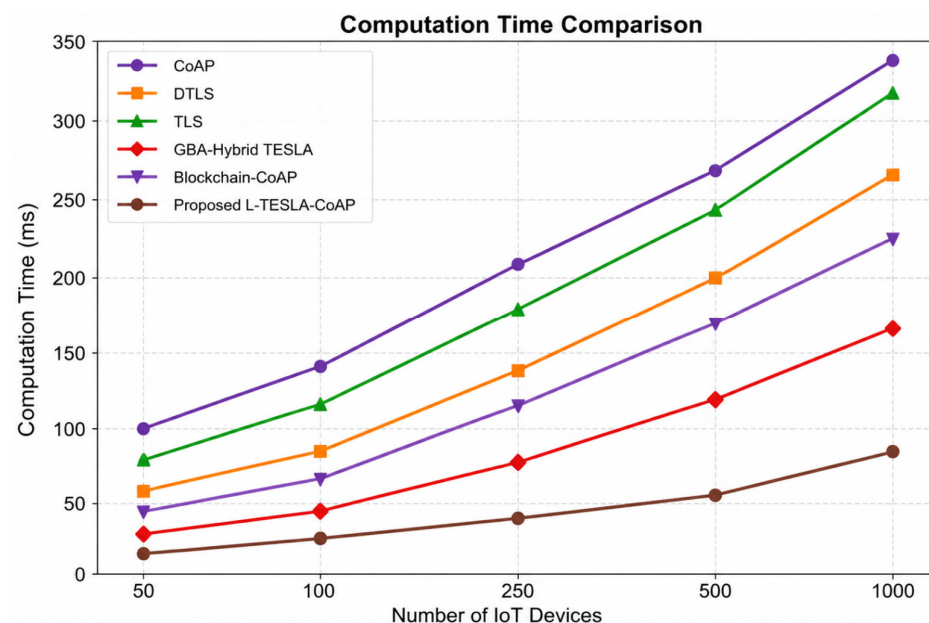


Figure 14. Comparison of the computation time of the proposed L-TESLA-CoAP framework with CoAP [9], DTLS [8], TLS [6], Blockchain-CoAP, and GBA-Hybrid TESLA under increasing numbers of IoT devices. Baseline methods are adapted from Refs. [7], [6], [5], [19], and [1], respectively.

7.8. Memory Consumption Analysis

Figure 15 illustrates the memory consumption of the proposed framework compared with CoAP [9], DTLS [8], TLS [6], GBA-Hybrid TESLA [36], and Blockchain-CoAP [11]. As expected, memory utilization increases with the number of connected IoT devices because additional authentication states and communication sessions must be maintained. Never-

theless, the proposed L-TESLA-CoAP framework consistently requires the smallest memory footprint among all compared approaches. Unlike blockchain-based authentication, which requires ledger maintenance, or TLS/DTLS-based schemes that maintain certificates and public-key parameters, the proposed framework stores only the TESLA key-chain state, lightweight SHA3-HMAC parameters, synchronization information, replay cache entries, and rotating pseudonym identifiers. This compact storage structure substantially reduces memory requirements, making the framework highly suitable for constrained IoT devices operating in edge-assisted 6G environments.

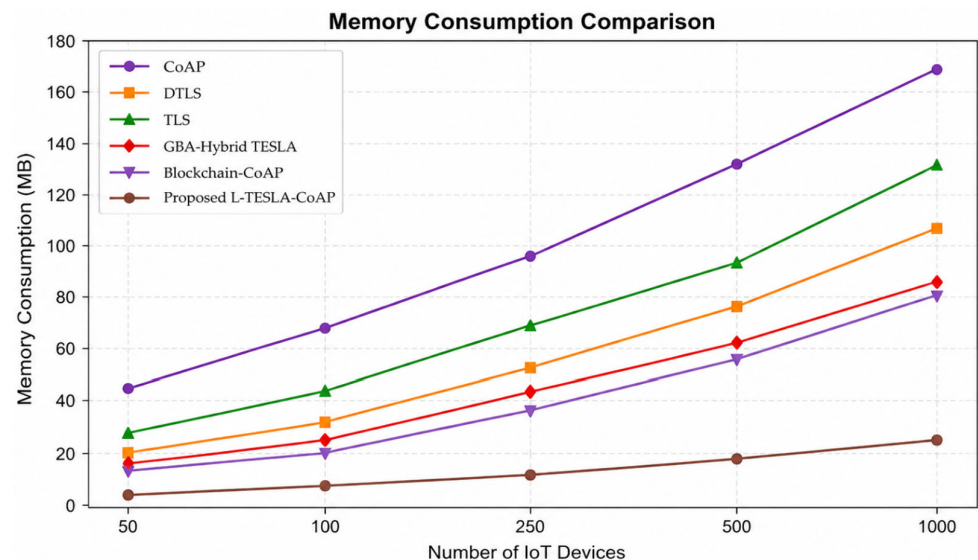


Figure 15. Comparison of the memory consumption (in MB) of the proposed L-TESLA-CoAP framework with CoAP, DTLS, TLS, Blockchain-CoAP, and GBA-Hybrid TESLA under increasing numbers of IoT devices. Baseline methods are adapted from Refs. [7], [6], [5], [19], and [1], respectively.

7.9. Authentication Success Rate Analysis

Authentication success rate measures the percentage of authentication requests that are successfully verified under varying network conditions. Figure 16 compares the authentication success rate of the proposed L-TESLA-CoAP framework with existing authentication schemes under increasing network load.

The proposed framework consistently achieves an authentication success rate exceeding 99% across all evaluated network sizes. This improvement results from the integration of adaptive synchronization, delayed TESLA key disclosure, replay-aware verification, and edge-assisted synchronization management, which together maintain synchronization continuity even under packet-loss conditions. In contrast, DTLS- and TLS-based schemes experience a gradual reduction in successful authentication due to repeated handshake failures and retransmission overhead. Blockchain-assisted authentication also results in lower authentication success rates because additional consensus delays increase synchronization failures under constrained network conditions. These results demonstrate that the proposed framework maintains reliable authentication while preserving lightweight operation for large-scale constrained 6G IoT environments.

7.10. Synchronization Error Rate vs. Number of IoT Devices

Synchronization error rate represents the percentage of synchronization attempts that fail because the synchronization delay exceeds the acceptable TESLA disclosure interval. Figure 17 presents the synchronization error rates obtained by different authentication frameworks.

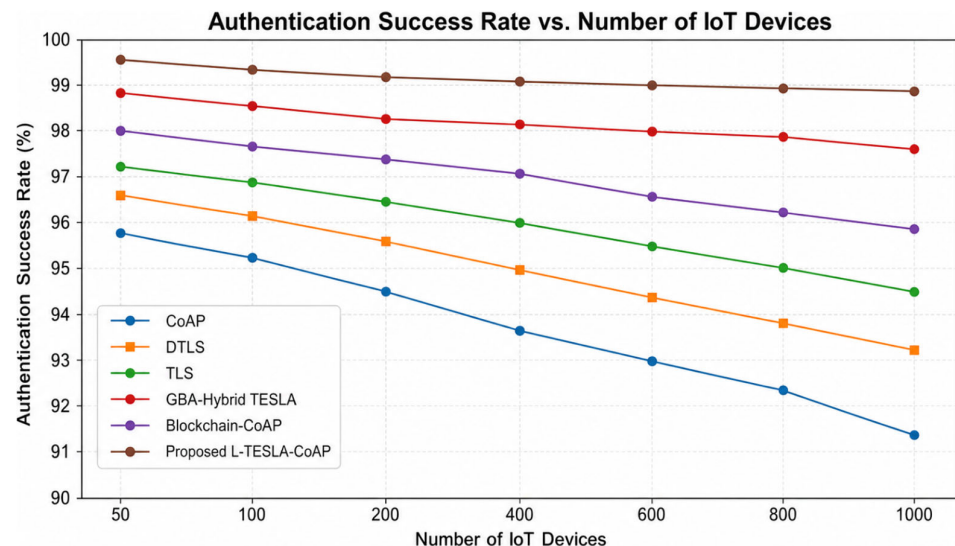


Figure 16. Comparison of the authentication success rate (%) of the proposed L-TESLA-CoAP framework with CoAP, DTLS, TLS, Blockchain-CoAP, and GBA-Hybrid TESLA under increasing numbers of IoT devices. Baseline methods are adapted from Refs. [7], [6], [5], [19], and [1], respectively.

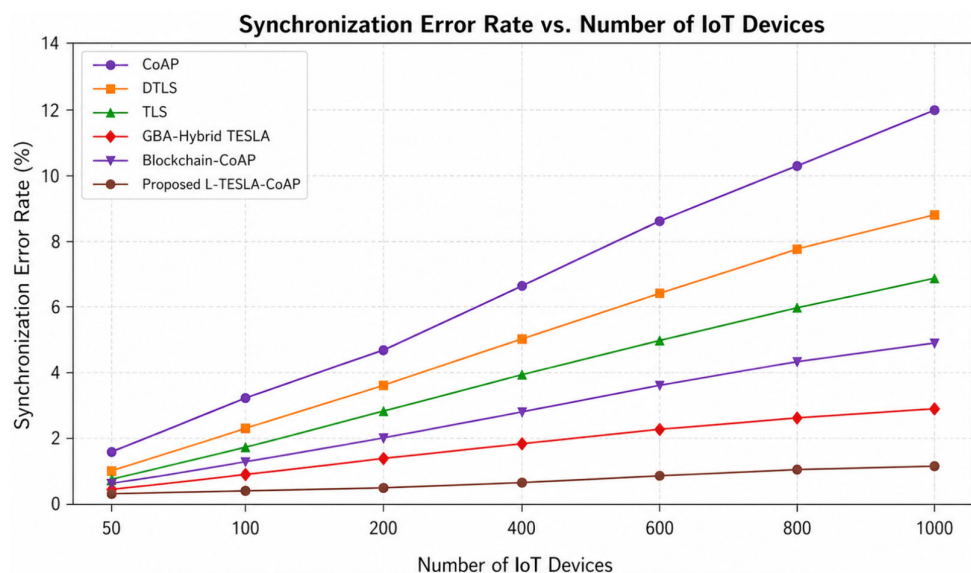


Figure 17. Comparison of the synchronization error rate (%) of the proposed L-TESLA-CoAP framework with CoAP, DTLS, TLS, Blockchain-CoAP, and GBA-Hybrid TESLA under increasing numbers of IoT devices. Baseline methods are adapted from Refs. [7], [6], [5], [19], and [1], respectively.

The proposed L-TESLA-CoAP framework exhibits the lowest synchronization error rate among all evaluated approaches. The adaptive synchronization controller continuously updates the TESLA disclosure interval based on packet-loss probability and network congestion, ensuring stable synchronization under dynamic wireless conditions. Consequently, synchronization errors remain below 1.5% throughout the experiments. Conventional TESLA implementations with fixed disclosure intervals exhibit higher synchronization errors under varying network conditions, whereas DTLS-, TLS-, and blockchain-based approaches are more sensitive to communication delays due to repeated negotiation and protocol overhead. These observations confirm the effectiveness of the adaptive synchronization mechanism in maintaining continuous authentication.

7.11. Storage Overhead Analysis

Storage overhead estimates the memory required to maintain authentication state information, including TESLA key-chain values, synchronization parameters, replay cache entries, and rotating pseudonym identifiers. Figure 18 compares the storage overhead of the evaluated authentication schemes.

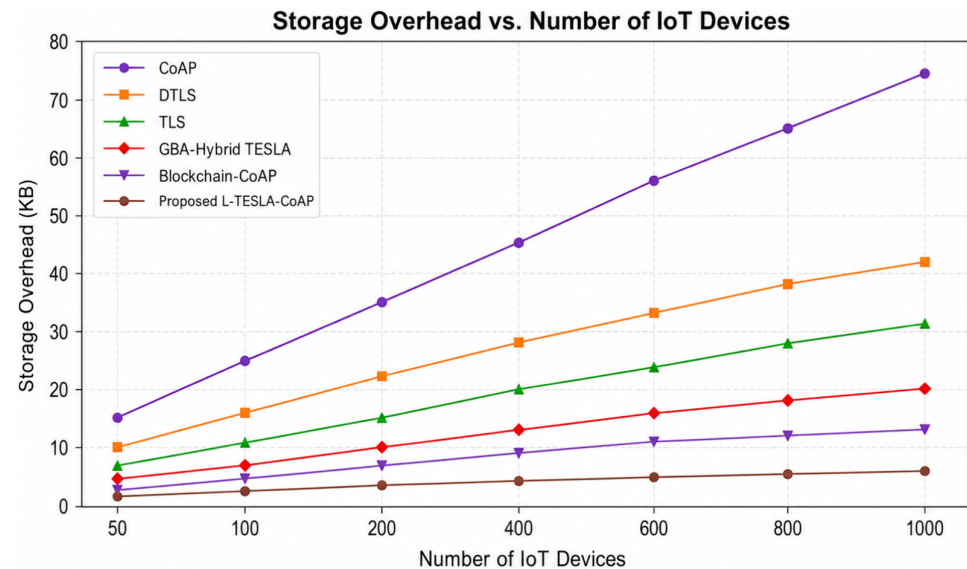


Figure 18. Comparison of the storage overhead (in KB) of the proposed L-TESLA-CoAP framework with CoAP, DTLS, TLS, Blockchain-CoAP, and GBA-Hybrid TESLA under increasing numbers of IoT devices. Baseline methods are adapted from Refs. [7], [6], [5], [19], and [1], respectively.

The proposed L-TESLA-CoAP framework maintains relatively low storage requirements by storing only the active TESLA hash-chain state, lightweight replay cache information, synchronization parameters, and current pseudonym identifiers. Unlike blockchain-assisted authentication schemes, which require maintaining distributed ledger information and synchronization metadata, the proposed framework avoids excessive storage consumption. Similarly, DTLS- and TLS-based approaches require additional session-state information and certificate-related data. The experimental results therefore demonstrate that the proposed framework provides an efficient balance between lightweight storage requirements and robust continuous authentication, making it well suited for memory-constrained IoT devices.

7.12. Security Attack Analysis

To further evaluate the robustness of the proposed authentication framework, Figure 19 compares the resistance of different authentication schemes to representative security attacks, including replay attacks, packet injection, denial-of-service (DoS), impersonation, and synchronization attacks. The proposed L-TESLA-CoAP framework consistently achieves the highest attack resistance across all evaluated attack categories. This superior performance is achieved through the integration of delayed TESLA key disclosure, SHA3-HMAC authentication, replay-cache management, adaptive synchronization, and rotating pseudonym identifiers. Replay attacks are effectively prevented through sequence-number validation and delayed key disclosure, while packet injection attacks are mitigated because attackers cannot generate valid authentication tags before key disclosure. Furthermore, adaptive synchronization improves resilience against synchronization manipulation and network congestion, whereas rotating pseudonyms reduce the risk of long-term device tracking and impersonation. Overall, the results demonstrate that the proposed framework provides stronger security protection than existing CoAP-, DTLS-,

TLS-, GBA-Hybrid TESLA-, and Blockchain-CoAP-based authentication schemes while maintaining lightweight operation suitable for dynamic 6G IoT networks.

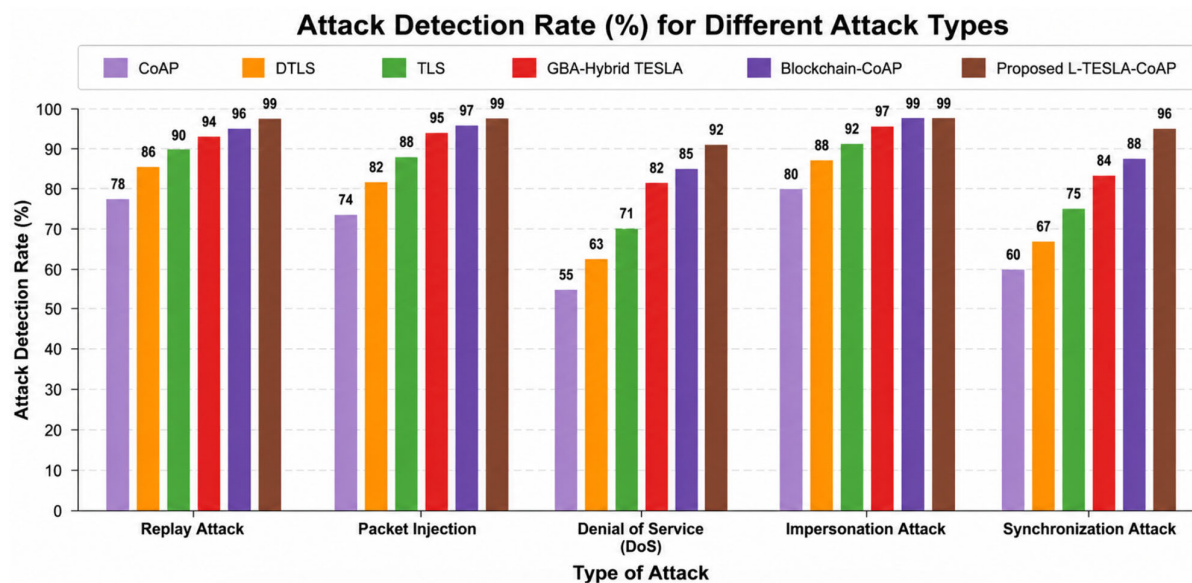


Figure 19. Comparison of the attack resistance (%) of the proposed L-TESLA-CoAP framework with CoAP, DTLS, TLS, Blockchain-CoAP, and GBA-Hybrid TESLA under different attack scenarios, including replay attacks, packet injection, denial-of-service (DoS), impersonation attacks, and synchronization attacks. Baseline methods are adapted from Refs. [7], [6], [5], [19], and [1], respectively.

7.13. Sensitivity Evaluation of Adaptive Synchronization Coefficients

To validate the selected adaptive synchronization parameters, a sensitivity evaluation was performed by varying the weighting coefficients α and β while maintaining all other simulation parameters unchanged. Figure 20 illustrates the influence of different coefficient combinations on authentication latency and packet-loss recovery. When smaller coefficient values are used, the adaptive synchronization controller responds more slowly to changes in packet loss and congestion, resulting in higher authentication latency and reduced synchronization robustness. The configuration $\alpha = 0.35$ and $\beta = 0.25$ achieves the best trade-off by minimizing authentication latency while maximizing packet-loss recovery. Therefore, these values were adopted throughout the experimental evaluation.

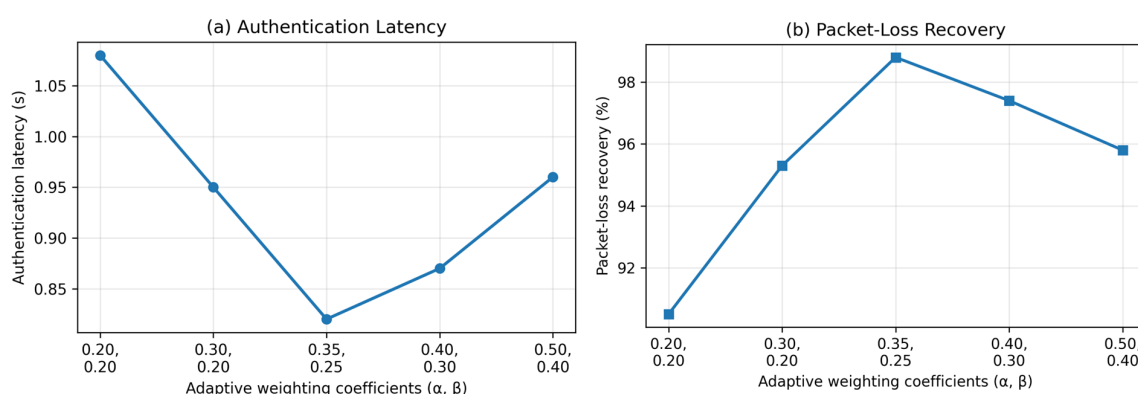


Figure 20. Sensitivity evaluation of the adaptive weighting coefficients: (a) authentication latency under different coefficient configurations and (b) packet-loss recovery under different coefficient configurations.

7.14. Ablation Study Analysis

The proposed L-TESLA-CoAP framework integrates several complementary mechanisms to achieve lightweight, scalable, and secure authentication for constrained 6G IoT environments. To assess the contribution of each design component, a qualitative ablation study was conducted by examining the expected impact of removing a single component while maintaining the remaining components unchanged. Unlike data-driven learning models, the proposed protocol is a deterministic authentication framework; therefore, the ablation study focuses on the functional contribution of each module to the overall system performance and security.

The analysis presented in Table 8 demonstrates that each component contributes to a specific aspect of the proposed framework. Adaptive synchronization primarily improves synchronization continuity and resilience to packet loss under dynamic network conditions. Edge-assisted synchronization reduces the computational burden on constrained IoT devices and enhances scalability by offloading synchronization management to nearby edge gateways. TESLA delayed key disclosure provides lightweight continuous authentication while improving tolerance to packet loss and preventing premature authentication forgery. Replay-aware synchronization enhances protection against replay attacks by verifying sequence numbers and managing the replay cache. Finally, the rotating pseudonym mechanism primarily enhances user privacy by reducing long-term device traceability without introducing additional communication overhead. Collectively, these mechanisms complement one another and enable the complete L-TESLA-CoAP framework to achieve a balanced combination of lightweight operation, scalability, synchronization stability, and robust security for edge-assisted 6G IoT environments.

Table 8. Qualitative Ablation Study of the Proposed L-TESLA-CoAP Framework.

Removed Component	Impact on Performance	Impact on Security
Adaptive Synchronization	Increases synchronization delay and reduces packet-loss recovery under dynamic wireless conditions.	Decreases synchronization robustness during congestion and intermittent connectivity.
Edge-Assisted Synchronization	Increases the computational burden on constrained IoT devices and reduces scalability.	Reduces synchronization efficiency and resilience in dense IoT deployments.
TESLA Delayed Key Disclosure	Increases authentication overhead and decreases packet-loss tolerance.	Weakens protection against packet forgery and replay attacks.
Replay-Aware Synchronization	Has little influence on communication latency but increases the probability of accepting duplicated packets.	Reduces resistance to replay attacks and synchronization-state manipulation.
Rotating Pseudonym Mechanism	Has negligible effect on communication performance.	Reduces privacy preservation and increases the risk of device tracking and impersonation.

8. Conclusions and Future Directions

This paper presented L-TESLA-CoAP, an edge-assisted lightweight authentication framework designed for constrained sixth-generation (6G) Internet of Things (IoT) environments. The proposed framework integrates Timed Efficient Stream Loss-Tolerant Authentication (TESLA) with the Constrained Application Protocol (CoAP) and introduces adaptive synchronization, rotating pseudonyms, replay protection, and edge-assisted verification to provide continuous, scalable, packet-level authentication. A comprehensive simulation-based evaluation demonstrated that the proposed framework consistently outperforms conventional CoAP-, DTLS-, TLS-, blockchain-, and GBA-based authentication approaches

in terms of authentication latency, energy consumption, packet overhead, throughput, packet-loss recovery, computation time, memory consumption, authentication success rate, synchronization accuracy, and storage efficiency. Additional sensitivity and ablation studies further verified the robustness and effectiveness of the proposed adaptive synchronization mechanism, as well as the contribution of each major framework component.

The proposed framework offers several practical advantages for resource-constrained 6G IoT deployments. By eliminating expensive public-key operations during continuous communication and leveraging lightweight hash-chain authentication with edge-assisted synchronization, L-TESLA-CoAP significantly reduces computational complexity while maintaining strong authentication guarantees. The integration of adaptive synchronization improves communication reliability under dynamic network conditions, whereas rotating pseudonyms enhance privacy protection without increasing protocol complexity. These characteristics make the proposed framework suitable for smart healthcare, industrial Internet of Things (IIoT), smart city infrastructures, intelligent transportation systems, environmental monitoring, and other large-scale edge-enabled IoT applications where low latency, energy efficiency, scalability, and continuous authentication are essential.

Future research will focus on implementing and validating the proposed framework on real constrained IoT hardware platforms, including ESP32, Raspberry Pi, Arduino, and edge gateway devices, to evaluate its performance under practical deployment conditions. In addition, future work will investigate mobility-aware synchronization for dynamic IoT environments, lightweight post-quantum authentication mechanisms, zero-trust edge architectures, and federated edge intelligence to further enhance scalability, resilience, and security in next-generation 6G IoT networks.

Funding: The Researchers would like to thank the Deanship of Graduate Studies and Scientific Research at Qassim University (<https://www.qu.edu.sa>) for financial support (QU-APC 2026).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Eledlebi, K.; Khoury, D.; Haddad, S.; Sondi, P.; Mun, H.; Danach, K.M.; Sayah, J.; Damiani, E.; Yeun, C.Y. Innovative Integration of GBA and Hybrid TESLA for Securing IoT Authentication in 5G and 6G Networks. *IEEE Access* **2025**, *13*, 176097–176109. [CrossRef]
2. Lee, L.-H.; Braud, T.; Zhou, P.; Wang, L.; Xu, D.; Lin, Z.; Kumar, A.; Bermejo, C.; Hui, P. All one needs to know about metaverse: A complete survey on technological singularity, virtual ecosystem, and research agenda. *arXiv* **2021**, arXiv:2110.05352.
3. Rizzardi, A.; Sicari, S.; Coen-Porisini, A. Analysis on functionalities and security features of Internet of Things related protocols. *Wirel. Netw.* **2022**, *28*, 2857–2887. [CrossRef]
4. Pereira, F.; Correia, R.; Pinho, P.; Lopes, S.I.; Carvalho, N.B. Challenges in resource-constrained IoT devices: Energy and communication as critical success factors for future IoT deployment. *Sensors* **2020**, *20*, 6420. [CrossRef] [PubMed]
5. Sheffer, Y.; Saint-Andre, P.; Fossati, T. Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS). IETF RFC9325. 2022. Available online: <https://www.rfc-editor.org/info/rfc9325/> (accessed on 7 July 2026).
6. Goworko, M.; Wytrębowicz, J. A secure communication system for constrained IoT devices—Experiences and recommendations. *Sensors* **2021**, *21*, 6906. [CrossRef] [PubMed]
7. Raza, S.; Shafagh, H.; Hewage, K.; Hummen, R.; Voigt, T. Lithe: Lightweight secure CoAP for the Internet of Things. *IEEE Sens. J.* **2013**, *13*, 3711–3720. [CrossRef]
8. Perrig, A.; Canetti, R.; Tygar, J.D.; Song, D. The TESLA broadcast authentication protocol. *CryptoBytes* **2002**, *5*, 2–13.
9. Eledlebi, K.; Alzubaidi, A.A.; Yeun, C.Y.; Damiani, E.; Mateu, V.; Al-Hammadi, Y. Simulation analysis and comparison of new hybrid TLI-μTESLA and variant TESLA protocols using SHA-2 and SHA-3 hash functions. *Sensors* **2022**, *22*, 9063. [CrossRef] [PubMed]

10. Eledlebi, K.; Alzubaidi, A.; Damiani, E.; Mateu, V.; Al-Hammadi, Y.; Puthal, D.; Yeun, C.Y. Comparative study of novel packet loss analysis and recovery capability between hybrid TLI- μ TESLA and other variant TESLA protocols. *Ad. Hoc. Netw.* **2024**, *163*, 103579. [\[CrossRef\]](#)
11. Eledlebi, K.; Alzubaidi, A.A.; Yeun, C.Y.; Damiani, E.; Mateu, V.; Al-Hammadi, Y. Enhanced inf-TESLA protocol: A continuous connectivity and low overhead authentication protocol via IoT devices. *IEEE Access* **2022**, *10*, 54912–54921. [\[CrossRef\]](#)
12. Al Dhaheri, A.; Yeun, C.Y.; Damiani, E. New two-level μ TESLA protocol for IoT environments. In Proceedings of the 2019 IEEE World Congress on Services (SERVICES), Milan, Italy, 8–13 July 2019; pp. 84–91.
13. Wu, Z.; Zhang, Y.; Liu, L.; Yue, M. TESLA-based authentication for BeiDou civil navigation message. *China Commun.* **2020**, *17*, 194–218. [\[CrossRef\]](#)
14. Chen, Y.; Kunz, T. Performance evaluation of IoT protocols under a constrained wireless access network. In Proceedings of the 2016 International Conference on Selected Topics in Mobile & Wireless Networking (MoWNeT), Cairo, Egypt, 11–13 April 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 1–7.
15. Garba, A.; Khoury, D.; Balian, P.; Haddad, S.; Sayah, J.; Chen, Z.; Guan, Z.; Hamdan, H.; Charafeddine, J.; Al-Mutib, K. LightCert4IoTs: Blockchain-based lightweight certificates authentication for IoT applications. *IEEE Access* **2023**, *11*, 28370–28383. [\[CrossRef\]](#)
16. Theodore, S.K.A.; Gandhi, K.R.; Palanisamy, V. A novel lightweight authentication and privacy-preserving protocol for vehicular ad hoc networks. *Complex Intell. Syst.* **2021**, *9*, 2981–2991. [\[CrossRef\]](#)
17. Henriques, M.S.; Vernekar, N.K. Using symmetric and asymmetric cryptography to secure communication between devices in IoT. In *Proceedings of the 2017 International Conference on IoT and Application (ICIOT 2017)*; IEEE: New York, NY, USA, 2017; pp. 1–4.
18. Taha, A.-E.-M.; Rashwan, A.M.; Hassanein, H.S. Secure communications for resource-constrained IoT devices. *Sensors* **2020**, *20*, 3637. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Khoury, D.; Haddad, S.; Sonidi, P.; Balian, P.; Harb, H.; Danach, K.; Merhej, J.; Sayah, J. CoAP/DTLS protocols in IoT based on blockchain light certificate. *IoT* **2025**, *6*, 4. [\[CrossRef\]](#)
20. Dutta, I.K.; Ghosh, B.; Bayoumi, M. Lightweight cryptography for Internet of Insecure Things: A survey. In Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 7–9 January 2019; pp. 475–481.
21. Gerodimos, A.; Maglaras, L.; Ferrag, M.A.; Ayres, N.; Kantzavelou, I. IoT: Communication protocols and security threats. *Internet Things Cyber-Phys. Syst.* **2023**, *3*, 1–13. [\[CrossRef\]](#)
22. Tawalbeh, L.A.; Somani, T.F. More secure Internet of Things using robust encryption algorithms against side channel attacks. In Proceedings of the IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA), Agadir, Morocco, 2 December 2016; pp. 1–6.
23. Sharbaf, M.S. IoT driving new business model, and IoT security, privacy, and awareness challenges. In Proceedings of the IEEE 8th World Forum on Internet of Things (WF-IoT), Yokohama, Japan, 26 October 2022; pp. 1–4.
24. Sethi, M.; Kortoci, P.; Di Francesco, M.; Aura, T. Secure and low-power authentication for resource-constrained devices. In Proceedings of the 2015 5th International Conference on the Internet of Things (IOT), Seoul, Republic of Korea, 26–28 October 2015; pp. 30–36.
25. Althobaiti, O.S.; Dohler, M. Cybersecurity challenges associated with the Internet of Things in a post-quantum world. *IEEE Access* **2020**, *8*, 157356–157381. [\[CrossRef\]](#)
26. Eledlebi, K.; Alzubaidi, A.; Damiani, E.; Puthal, D.; Mateu, V.; Zemerly, M.J.; Al-Hammadi, Y.; Yeun, C.Y. Bio-integrated hybrid TESLA: A fully symmetric lightweight authentication protocol. *IEEE Internet Things J.* **2024**, *11*, 37421–37436. [\[CrossRef\]](#)
27. Wu, T.-Y.; Meng, Q.; Kumari, S.; Zhang, P. Rotating behind security: A lightweight authentication protocol based on IoT-enabled cloud computing environments. *Sensors* **2022**, *22*, 3858. [\[CrossRef\]](#) [\[PubMed\]](#)
28. Rao, P.M.; Deebak, B.D. A comprehensive survey on authentication and secure key management in Internet of Things: Challenges, countermeasures, and future directions. *Ad. Hoc. Netw.* **2023**, *146*, 103159. [\[CrossRef\]](#)
29. Rana, M.; Mamun, Q.; Islam, R. Enhancing IoT security: An innovative key management system for lightweight block ciphers. *Sensors* **2023**, *23*, 7678. [\[CrossRef\]](#) [\[PubMed\]](#)
30. Garcia-Carrillo, D.; Marin-Lopez, R. Lightweight CoAP-based bootstrapping service for the Internet of Things. *Sensors* **2016**, *16*, 358. [\[CrossRef\]](#) [\[PubMed\]](#)
31. Khushu, A.; Zgonjanin, D.; Kim, N. *Generic Bootstrapping Architecture (GBA) Based Security over Constrained Application Protocol (CoAP) for IoT Devices*; Ericsson White Paper: Stockholm, Sweden, 2019.
32. Sethi, M.; Lehtovirta, V.; Salmela, P. Using Generic Bootstrapping Architecture with Constrained Devices. *Internet-Draft*. **2014**, 1–18. Available online: <https://datatracker.ietf.org/doc/draft-sethi-gba-constrained/00/> (accessed on 7 July 2026).
33. Mun, H.; Damiani, E.; Yeun, C.Y. Edge-assisted lightweight authentication for ultra-massive IoT in 6G environments. *IEEE Access* **2024**, *Early access*.

34. Turnip, T.N.; Andersen, B.; Vargas-Rosales, C. Towards 6G Authentication and Key Agreement Protocol: A Survey on Hybrid Post-Quantum Cryptography. *IEEE Commun. Surv. Tutor.* **2025**, *28*, 3311–3345.
35. Bensari, M.; Bilami, A.; Bilami, K.E.; Lorenz, P.; Gaber, J. Lightweight Dual Blockchain Authentication for 6G-Enabled IoT Environments. *Future Internet* **2026**, *7*, 64. [[CrossRef](#)]
36. Alnaim, A.K.; Alwakeel, A.M. Zero-Trust Mechanisms for Securing Distributed Edge and Fog Computing in 6G Networks. *Mathematics* **2025**, *13*, 1239. [[CrossRef](#)]
37. Faleh, M.A.; Abdulsada, A.M.; Alaidany, A.A.; Al-Shareeda, M.A.; Almaiah, M.A.; Shehab, R. SECRE-MEN: A Lightweight Quantum-Resilient Authentication Framework for MEC-Enabled 6G-IoT Networks. *J. Robot. Control* **2025**, *6*, 1681–1692. [[CrossRef](#)]
38. Kudaloor, A.; Aijaz, A. *Toward Quantum-Safe 6G: Experimental Evaluation of Post-Quantum Cryptography for Resource-Constrained Networks*; IEEE: New York, NY, USA, 2026.
39. Ma, W.; Zhu, L.; Tan, Y.; Zheng, B.; Zhang, Y.; Zhang, Y.; Ying, K.; Gao, Z.; Sun, H.; Shao, X.; et al. A Survey on Reconfigurable and Movable Antennas for Wireless Communications and Sensing. *IEEE Commun. Surv. Tutor.* **2026**, *28*, 4842–4882. [[CrossRef](#)]
40. Tefera, M.A.; Yao, C.-K.; Peng, C.-H.; Chung, Y.-J.; Lin, T.-C.; Chen, H.-M.; Hsu, W.-Y.; Peng, P.-C. Optimizing Free-Space Optical Communication Based Distributed Temperature Sensing Demodulation Using Meta-Learning. *J. Light. Technol.* **2025**, *43*, 10032–10039. [[CrossRef](#)]
41. 3GPP. Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA). 3GPP TS 33.220. Release 6. 2024. Available online: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2280> (accessed on 7 July 2026).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.