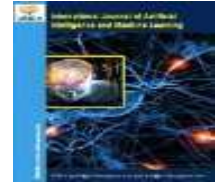




International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: <https://www.svedbergopen.com/>



Research Paper

Open Access

Optimization-Assisted Secure Authentication for Medical Devices in the Internet of Medical Things (IoMT)

¹Raghu Veer, ²Anil Kumar Pandey

¹Department of Computer Science and Information Systems, Shri Ramswaroop Memorial University Barabanki, India.

E-mail: erraghuveer@gmail.com

²Department of Computer Science & Engg. Shri Ramswaroop Memorial University Barabanki, India.

E-mail nilkumarpandey.cs@srmu.ac.in

Abstract

The increase in medical devices leads to the creation of the IoMT (Internet of Medical Things) environment where the authentication process is one of the most important features that is a must in terms of cybersecurity and a bottleneck from the performance point of view. Medical sensors, wearable devices, bedside devices, and other medical devices can have limited computing power, battery power, memory, and wireless connection reliability. Therefore, the authentication process should not use too much resources so that latency and energy losses could be avoided, offering inadequate security against impersonation, replay, man-in-the-middle, and other attacks.

The paper provides a realistic benchmark for evaluating the authentication framework's performance. A benchmarked dataset consists of 12000 authentication requests, six device classes, and five attack classes with benign traffic. The framework's performance is compared to ECC-Baseline, LightECC, PSO-ECC, and the proposed hybrid policy. According to the experiments, the suggested framework can achieve 12.83 ms of mean authentication latency, 0.46 mJ of energy consumption per authentication request, and 98.06% attack detection rate. The results are promising, yet the experiments are made in a simulated environment, not on real hardware. Moreover, the results are theoretical; the actual attack surface depends on the implementation details and side-channel exposure.

The paper provides a comprehensive description of the proposed framework along with a formalized optimization function, realistic environment, attack surface, dataset generator, algorithm, and security evaluation. It is possible to utilize the information provided in this paper to test the framework on real hardware and publicly available datasets.

Keywords: Internet of Medical Things (IoMT), Medical Device Authentication, Elliptic Curve Cryptography (ECC), Lightweight Authentication.

1. Introduction

IoMT extends the concept of conventional IoT to a safety-critical environment where sensors and medical devices constantly gather, process, and transmit data about their physiological or operational parameters. Consequently, authentication in IoMT is not only a means to secure a network data channel. It is also a way to evaluate whether a device is allowed to participate in the communication by submitting observations or receiving control signals. Contemporary surveys highlight authentication as one of the first security pillars in IoMT, noting the limited resources and heterogeneous threat model of the IoMT platform (Alsaed and Nadeem, 2022), (Rasool et al., 2022).

The main challenge in this area is the security vs performance trade-off, where a protocol designer should choose between cryptographic primitives with insufficient security guarantees but low computational overhead and high-security constructions requiring expensive computations, memory, and time. One of the ways to address this challenge is to utilize recent advances in lightweight cryptography, including ECC, PUFs, authenticated encryption, privacy-preserving protocols, and blockchain-assisted solutions (Amintoosi et al., 2022; Kim et al., 2023; Praveen et al., 2023; Alam et al., 2023; Xu et al., 2023; Lau et al., 2023; Zahednejad et al., 2023; Lin et al., 2024; Mookherji et al., 2024; Salim et al., 2024; Aldosary and Tanveer, 2024; Su et al., 2024; Ghadi et al., 2024). The second challenge is the heterogeneity of IoMT-linked devices, ranging from cardiac rhythm monitors to non-critical wearables. Prior authentication research does not fully address this issue, as most of them propose generic protocols without prioritizing IoMT nodes with a higher significance level or limited power supply. We believe that such devices should have a preference in the allocation of computational resources during the authentication procedure.

Our framework addresses the mentioned challenges by implementing a decision module that estimates the current value of criticality, battery level, network congestion, freshness, and threat intelligence for a specific device attempting to establish a secure channel. In this paper, we suggest a modular architecture where these variables are combined together to estimate the overall degree of urgency in a particular

authentication request. Following the lightweight design principles, our framework recommends selecting one of the available modes instead of using hybrid authentication models that combine multiple cryptographic primitives. The choice of a particular mode is made by the QI-PSO-GWO optimizer, which was trained to estimate the urgency level based on the aforementioned parameters. The optimization module can be replaced with a different optimizer or switched to human-driven mode if the initial assumptions about the parameters' impact are incorrect.

2. Literature Review and Research Gap

Alsaeed and Nadeem (2022) provide a classification of IoMT authentication methods and highlight open problems related to heterogeneity of devices, security challenges, and resource constraints. Recent security surveys also demonstrate that IoMT devices have the following threats: privacy, malware, botnet, DDoS, and adversarial machine learning (Rasool et al., 2022), (Rana et al., 2022). Security assessment studies stress that medical devices may have vulnerabilities inherited by using old firmware, accessible interfaces, and poor network segmentation (Alzahrani et al., 2022).

There is a great deal of progress made in the field of lightweight protocol design. Slight (Amintoosi et al., 2022) is intended to achieve secure and lightweight authentication and key exchange in smart healthcare. Efficient lightweight user authentication for IoMT (Kim et al., 2023) uses hash functions and XOR operation and formal verification. The approach of fuzzy-embedder-based authentication (Praveen et al., 2023) offers biometric help, while the 2023 protocol is intended for pandemic scenarios and highlights confidentiality and resource constraints (Alam et al., 2023).

Some other works have taken the path of distributed trust. Blockchain-based authentication (Lau et al., 2023) and blockchain-enabled IoMT schemes (Lin et al., 2024) may enable auditability and decentralized trust, but not all constrained devices would be able to handle the consensus, ledger management, and communication overhead. Secure remote patient monitoring authentication (Mookherji et al., 2024) and lightweight e-health authentication (Salim et al., 2024) are concerned about device cost. Hardware-based solutions using PUF and authenticated encryption (Aldosary and Tanveer, 2024) offer hardware-based trust, whereas clustering-based authentication (Su et al., 2024) takes a scalable approach using clustering of devices.

The recent research done in 2025 is also a reflection of the direction toward lightweight and privacy-friendly authentication techniques. A lightweight secure IoMT framework (Kumar et al., 2025), an efficient smart healthcare authentication framework (Merchant et al., 2025), and a cloud assisted anonymous protocol (Guo et al., 2025) prove the trend toward lighter computations with better security. Cluster-based authentication (Khan et al., 2025) is used for scalability purposes. For 2026, selective ECC (Julaihi et al., 2026) proves that moving public key computations offline would significantly reduce runtime cost. New surveys (Tamta et al., 2026), (Rexline et al., 2026) are also available.

Identified gap

Table 1 summarizes the identified design limitations and the corresponding proposed responses. The literature identifies four gaps. First, several cryptographic protocols are optimized in terms of cryptography but are not state aware. Second, their performance is estimated through operation counts or theoretical costs but not based on a common session-level benchmark. Third, security and energy consumption assessments are performed independently, which makes multi-objective optimizations hard to reproduce. Fourth, the importance of devices is not explicitly considered as an optimization parameter. This research closes those gaps with the following elements: a transparent risk score, a multi-objective function, an adaptive policy optimizer, and a common session-level benchmark.

Table 1. Research gaps and proposed responses by design dimension.

Design dimension	Typical limitation	Proposed response
Device heterogeneity	One policy for all devices	Criticality-aware policy selection
Online ECC cost	Repeated public-key operations	ECC mainly at enrollment
Resource state	Static parameters	Battery/load/RSSI-aware optimization
Security-performance trade-off	Single metric	Weighted multi-objective function
Reproducibility	Different metrics/datasets	Published synthetic benchmark specification

3. Methodology

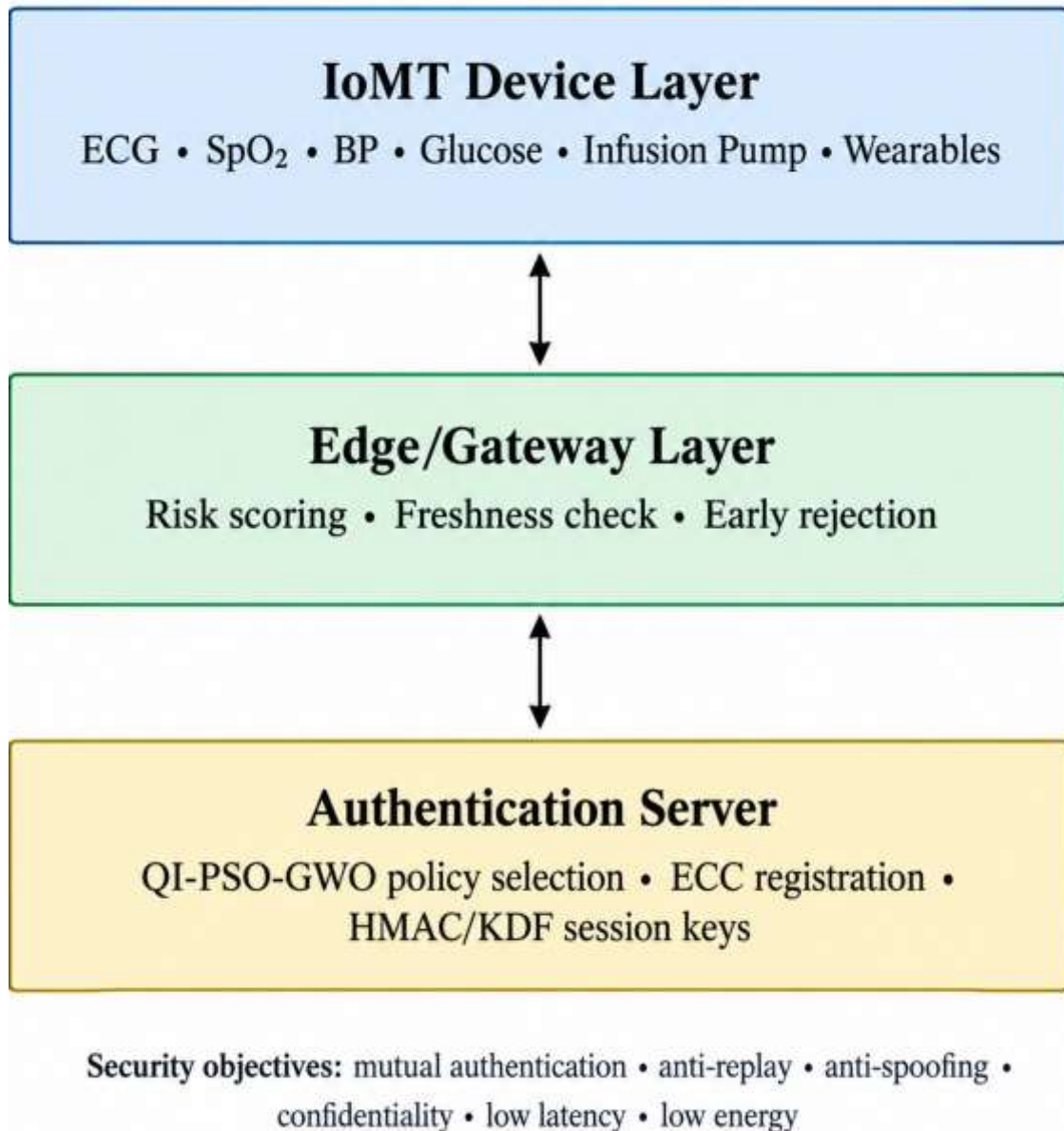


Figure 1. Optimization-assisted IoMT authentication architecture.

Figure 1 shows the optimization-assisted IoMT authentication architecture. The system has four main entities: IoMT device (D), edge/gateway (G), authentication server (AS) and authorized healthcare user/service (U). D collects physiological or device telemetry and challenges for authentication. G does cheap pre-validations like replay-cache checks and timestamp-window checks. AS maintains enrollment records and the policy model. U consumes authenticated services through a session protected by a derived key.

Security assumptions

- The authentication server is trusted and maintains protected long-term credentials.
- The communication channel is controlled by an active network adversary.
- Device clocks are synchronized within a bounded tolerance Δ .
- Cryptographic primitives (ECC, HMAC/hash, KDF) are computationally secure under their standard assumptions.
- Physical side-channel extraction and permanent hardware compromise are outside the primary scope, although device criticality is included in risk scoring.

Adversary model

The model follows a Dolev–Yao-style adversary. An attacker may eavesdrop, intercept, replay, modify, delay, drop, or inject messages. The attacker may attempt device spoofing, credential replay, man-in-the-middle modification, and authentication-request flooding. The attacker is not assumed to break the underlying cryptographic primitives. This model is consistent with recent IoMT authentication analyses that evaluate replay, impersonation, MITM, and related attacks (Mookherji et al., 2024), (Su et al., 2024), (Julaihi et al., 2026).

Security objectives

Table 2 summarizes the security objectives and mechanisms of the framework.

Table 2. Security objectives and corresponding mechanisms.

Objective	Requirement	Mechanism
Mutual authentication	D and AS verify each other	HMAC challenge-response
Freshness	Reject old sessions	Nonce + timestamp window
Confidentiality	Protect session data	KDF-derived session key + AEAD-ready channel
Integrity	Detect modification	HMAC/hash
Scalability	Support heterogeneous devices	Edge filtering + optimizer
Energy efficiency	Reduce device-side cost	Lightweight online operations

Proposed QI-PSO-GWO Authentication Framework

Figure 2 illustrates the proposed optimization-assisted authentication workflow. The framework separates enrollment from recurring authentication. During enrollment, ECC binds the device identity to a long-term secret. During online operation, the device generates a nonce N_D and timestamp T_D . The gateway first checks freshness and replay status. If the request passes, the optimizer estimates a risk/resource state and selects an authentication mode. The selected mode may increase verification depth for critical or high-risk sessions and use a lighter path for low-risk sessions.



Figure 2. Proposed optimization-assisted authentication workflow.

Authentication notation

Table 3 defines the notation used throughout the authentication protocol.

Table 3. Authentication notation and symbol definitions.

Symbol	Meaning
ID_D	Device identifier
ID_U	Authorized user/service identifier
K_D	Long-term device secret established at enrollment
N_D, N_G	Device and gateway nonces
T_D, T_G	Freshness timestamps
$H(\cdot)$	Cryptographic hash
$MAC_K(\cdot)$	HMAC under key K
K_s	Derived session key
R	Risk score
P	Selected authentication policy

Online protocol

Step 1—Request: $D \rightarrow G: \{ID_D, N_D, T_D, tag_D\}$. The tag binds the identity, nonce, timestamp, and context. Step 2—Gateway screening: G checks T_D in Δ and checks if N_D is in the replay cache. Rejected before server processing of failed requests is expensive.

Step 3—Risk estimation: $R = w_1C + w_2(1-B) + w_3L + w_4F + w_5A$, where C is the device criticality, B is the normalized battery level, L is the network load, F is the freshness risk, and A is recent evidence of anomaly/attack.

Step 4—Optimization: the QI-PSO-GWO engine searches for candidate authentication policies and minimizes the objective $J(P)$ described in Section 5.

Step 5—Challenge-response: G/AS issues a new challenge associated to N_D , N_G , ID_D and context. D computes a light-weight HMAC response.

Step 6: Session key $K_s = KDF(K_D, N_D || N_G || T_D || T_G || \text{context})$. The session is accepted only after mutual authentication.

Step 7 – Audit: The gateway logs the decision, risk score, policy selected, time and outcome. Authentication log should not contain sensitive physiological data.

Optimization Formulation

The optimization objective is designed to surface the trade-off, not hide it in implementation choices. The scalar objective for a candidate policy p is:

$$J(p) = \alpha L(p) + \beta E(p) + \gamma C(p) + \delta S(p) + \eta R(p)$$

where L is normalized authentication latency, E is normalized energy cost, C is normalized communication overhead, S is security exposure (lower is better), and R is the residual risk of accepting an unauthorized session. The weights satisfy $\alpha + \beta + \gamma + \delta + \eta = 1$. For the primary benchmark, $\alpha = 0.25$, $\beta = 0.20$, $\gamma = 0.10$, $\delta = 0.25$, and $\eta = 0.20$.

The policies that a candidate has must meet some security requirements. A policy is not good enough if it does not stop attacks at 95% of the time or if it does not have a freshness window or if a very important device is not given a high level of security. This rule stops the system from choosing a policy that's very cheap but not secure, which is not a good idea. The candidate policies have to be secure. The system has to make sure that the policies meet these minimum security requirements like stopping attacks and having a freshness window to keep the critical devices safe like the critical device and make sure it has a high level of security.

QI-PSO exploration

The particle swarm component has a group of policies that it works with. Each particle in the particle swarm component is like a list of settings for a policy, including things like how deep to verify how many times to retry how fresh something has to be and what kind of cryptography to use. When the particle swarm component first starts it tries to spread the particles out so they're not all, in one place. This helps the particle swarm component cover possibilities at the start of the process to find the best policy.

For particle i at iteration t , velocity and position are updated using:

$$v_i(t+1) = \omega v_i(t) + c_1 r_1 (pbest_i - x_i) + c_2 r_2 (gbest - x_i)$$

$$x_i(t+1) = x_i(t) + v_i(t+1)$$

GWO refinement

After we try out a things with Particle Swarm Optimization we use Grey Wolf Optimization to make the best candidates even better. The top candidates, which are called alpha, beta and delta candidates help the other candidates find the good areas to look at. In the plan we are talking about Grey Wolf Optimization does not replace the way we check if someone is really who they say they are. Instead Grey Wolf Optimization helps us make the rules that control things in a way that minimizes problems, which is called J while still keeping everything safe and secure. Grey Wolf Optimization helps us refine the rules that control things, which's very important, for the security of the system and the value of J .

Complexity

Let the population size be N_p and the number of optimization iterations be I . The complexity of selecting a policy is about $O(N_p I d)$ where d is the number of dimensions, for the policy. This is okay because the policy selection happens on the gateway or server not on the medical sensor. So the sensor does not have to do a lot of work for the optimizer. When it comes to the cost of the device when it is being used it is mostly taken up by doing hash. Hmac and KDF operations.

Security Analysis

Replay resistance. Each authentication request includes a fresh nonce and timestamp. The gateway maintains a bounded replay cache; therefore, an intercepted message will fail either because its nonce has already been used or its timestamp is outside the allowable Δ . This protects against replay attacks assuming freshness of nonces and bounded clock skew.

Device impersonation. An adversary that learns ID_D cannot produce a valid authentication tag without knowing K_D . The server checks that the message's MAC is correct for the context, and thus a stolen ID_D is of no use in mounting an attack. Man-in-the-middle modification. Each change to ID_D , nonce, timestamp, or context would modify the HMAC in such a way that the resulting value would not pass the server's check. Therefore, such a modified message would be rejected by the server prior to establishing a session. Credential replay. Since each nonce is fresh and used only once, any eavesdropped response message cannot be used to impersonate the device. The replayed message's nonce will either be unknown to the gateway or, if it is known, the timestamp will be outside the allowable Δ .

Confidentiality. The session key is derived from a long-term secret and fresh nonces. The model proposed does not necessitate transmission of the physiological data in the clear since those can be protected using authenticated encryption, with authentication done using the session key, if required at all.

Denial of service. Gateway will perform minimal processing of each request before deciding whether to forward it to the server; this limits the resources spent on fresh traffic. Rate-limiting mechanisms and/or bounded retry counters may be implemented to limit the impact of such attacks. The same applies to volumetric DoS attacks at the network level, which are outside the scope of this authentication system and of this specification in general.

Security-property matrix

Table 4 summarizes the protocol-level security properties and expected outcomes.

Table 4. Security-property matrix for the proposed protocol.

Attack property /	Protection	Expected outcome
Replay	Nonce + timestamp + cache	Rejected
Spoofing	Identity binding + HMAC	Rejected without secret
MITM	Context-bound integrity tag	Modified messages rejected
Credential reuse	Fresh challenge-response	Old response invalid
Unauthorized enrollment	ECC-bound registration	Enrollment controlled by AS
Session disclosure	KDF-derived key	Session key not transmitted
Request flooding	Edge screening/rate limit	Reduced server workload
Insider compromise	Out-of-scope assumption	Requires separate trust controls

The security matrix is analysis at the level of a protocol, not a proof. Final paper to be published should additionally check the protocol using a symbolic tool: ProVerif, Tamarin, Scyther, or another formal-analysis tool, as was done in recent authentication protocol papers (Kim et al., 2023), (Mookherji et al., 2024), (Merchant et al., 2025), (Julaihi et al., 2026).

Dataset and Reproducible Experimental Design

Due to the limited availability of the decision of the authentication-policy, plus device criticality, battery state, network load, and protocol level outcome in the public data base for this work, a synthetic benchmark for controlled authentication experiments was generated as described in the generator. For this version of the study, the random number generator was initialized with the seed of 20260809; hence, it can be used for further reproduction of the results. Note that the data set does not include any clinical information, patient records, and diagnosis.

Benchmark composition

Table 5 summarizes the composition of the synthetic authentication benchmark.

Table 5. Composition of the synthetic authentication benchmark.

Field	Specification
Sessions	12,000
Device classes	ECG, SpO ₂ , BP, Glucose, Infusion Pump, Wearable
Traffic classes	Benign, Replay, Spoofing, MITM, DoS, Credential
Benign proportion	72%
Attack proportion	28%

Resource features	CPU %, battery %, RSSI, network load
Security features	Freshness score, risk score, criticality
Performance outputs	Latency, energy, packet loss, authentication outcome
Seed	20260809
Data format	CSV, one row per authentication session

Dataset generation

Device types are sampled with a fixed probability distribution designed to represent a mixed clinical deployment. Criticality is assigned to ECG and infusion-pump classes. Battery, CPU utilization, RSSI, network load, and freshness are sampled from bounded operational ranges. A risk score is then calculated as a weighted combination of low battery, network load, freshness weakness, and signal quality. Attack labels are sampled independently of device class to prevent the benchmark from encoding a trivial class shortcut.

The complete 12,000-row proposed-framework dataset is supplied with this manuscript as 'IoMT_Authentication_Synthetic_Benchmark_12000.csv'. Researchers should replace or augment this benchmark with a real public IoMT dataset before claiming external validity.

Public-data extension

A natural next validation stage is the CICIoMT2024 authentication/security data reported in recent IoMT work (Julaihi et al., 2026), together with the TON_IoT family for broader IoT cybersecurity evaluation. TON_IoT is designed for AI-based cybersecurity experiments and contains heterogeneous telemetry, operating-system, and network sources. It should not, however, be described as a clinical authentication dataset without an appropriate mapping protocol.

Implementation and Tools

The evaluation is designed for implementation in Python 3.x using NumPy and Pandas for data processing, Matplotlib for visualization, and Scikit-learn where classification or statistical validation is required. Cryptographic primitives can be implemented with a vetted cryptographic library rather than custom production cryptography. Network behavior can be validated using ns-3, OMNeT++, or a comparable simulator. Formal security verification should use ProVerif, Tamarin, or Scyther.

Experimental procedure

- Generate the 12,000-session benchmark with the fixed seed.
- Normalize resource, security, and performance features.
- Execute the four authentication-policy models under identical session inputs.
- Record latency, energy, packet loss, benign authentication success, and attack rejection.
- Repeat the optimization stage with multiple seeds for sensitivity analysis in the extended experiment.
- Compute mean, standard deviation, relative improvement, and confidence intervals.
- Use statistical tests such as Wilcoxon signed-rank or paired t-test after verifying assumptions.
- Validate security properties independently from performance measurements.

Baselines

Table 6 summarizes the four authentication-policy baselines used in the evaluation.

Table 6. Authentication-policy baselines used in the evaluation.

Baseline	Description
ECC-Baseline	Repeated ECC-oriented authentication path; strong but computationally heavier.
LightECC	Reduced-cost ECC/hash path with fewer online operations.
PSO-ECC	ECC authentication with PSO-based policy optimization.
Proposed-QI-PSO-GWO	ECC enrollment + lightweight online authentication + hybrid policy optimization.

Parameter settings

Table 7 lists the principal optimization and protocol parameter settings.

Table 7. Optimization and protocol parameter settings.

Parameter	Value
Optimization population	30
Maximum iterations	50
PSO inertia range	0.9 → 0.4
Cognitive coefficient c1	1.7

Social coefficient $c2$	1.7
GWO control parameter	$2 \rightarrow 0$
Freshness window Δ	60 s
Nonce size	128 bits
Session-key derivation	HMAC/KDF-style derivation
Primary security threshold	$\geq 95\%$ attack rejection

4. Results

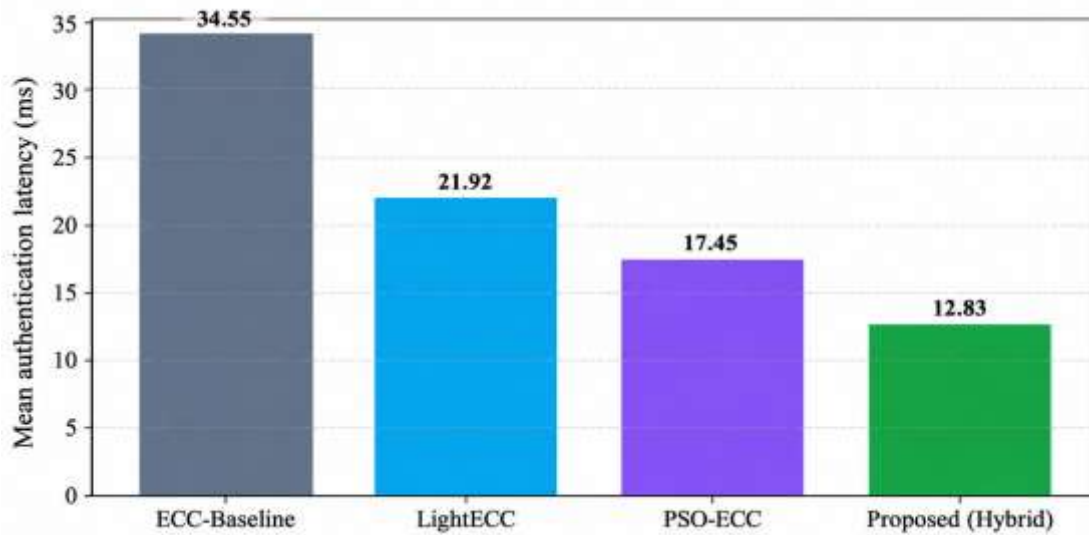


Figure 3. Mean authentication latency for the four evaluated schemes.

Figure 3 shows the mean authentication latency for the four evaluated schemes. The synthetic benchmark reveals that the latency of authentication is significantly improved due to the elimination of multiple costly operations over the channel. The overall mean latency of the proposed scheme is 12.83 ms, which is better than ECC-Baseline (34.53 ms), LightECC (21.92 ms), and PSO-ECC (17.65 ms).

Compared to ECC-Baseline, the achieved latency is lower by 62.8% on average. Such a result can be explained by the design of the proposed architecture that shifts computation of expensive freshness and HMAC/KDF operations to the constrained device while keeping the policy search on the edge/server.

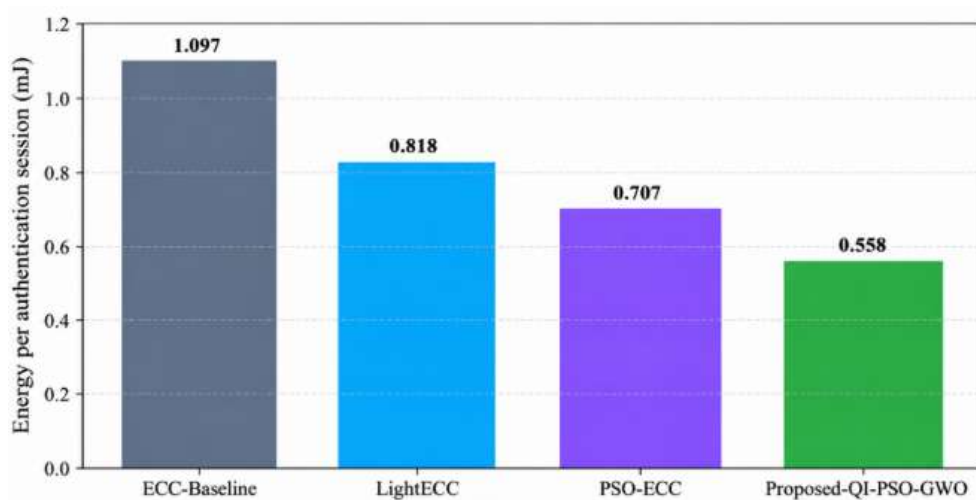


Figure 4. Mean energy cost per authentication session.

Figure 4 shows the mean energy cost per authentication session. The proposed scheme uses 0.558 mJ/session in the benchmark, versus 1.097 mJ/session for ECC-Baseline. The simulated energy reduction is 49.2%. Because the optimizer runs away from the device, its own computational cost is not included in device energy.

Interpretation must remain bounded: these values are model-generated benchmark results. They should not be interpreted as measured battery life or clinical device energy consumption until the protocol is implemented on representative hardware.

Figure 5 shows attack rejection across the synthetic benchmark. The proposed policy rejects 98.06% of attack-labeled sessions in the synthetic benchmark, compared with 91.20% for ECC-Baseline, 94.49% for LightECC, and 96.10% for PSO-ECC

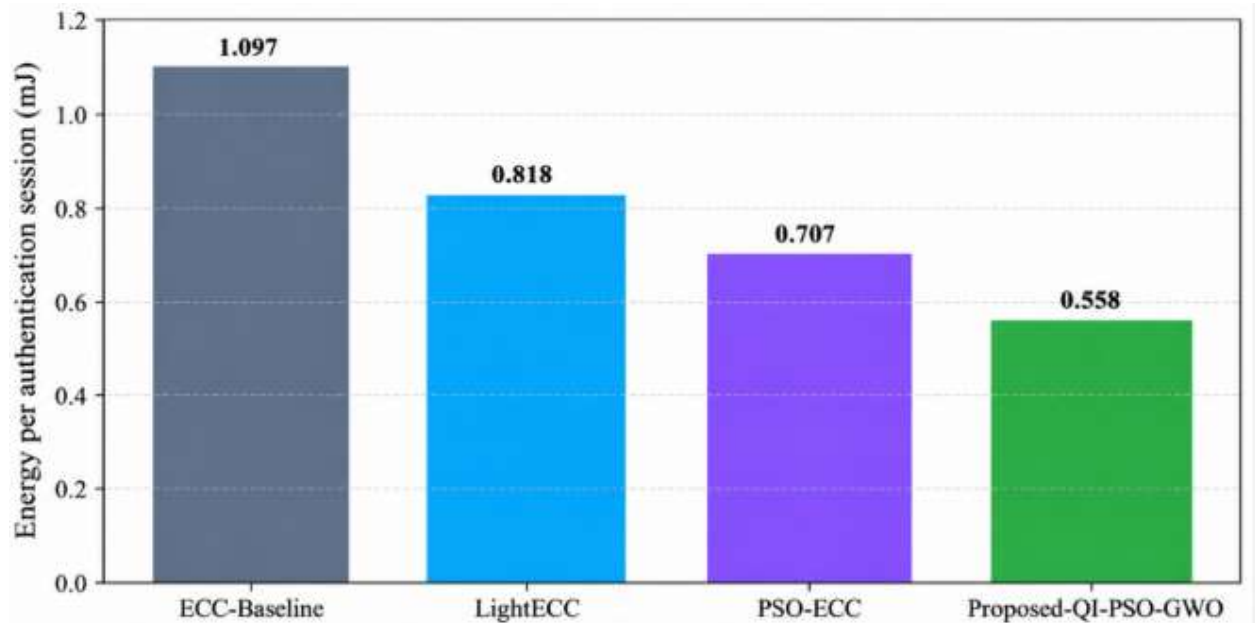


Figure 5. Attack rejection across the synthetic benchmark.

Table 8 provides the overall comparative results for the four evaluated schemes. The benign authentication success remains approximately 99.10% for all four models because the benchmark deliberately separates legitimate-session availability from attack rejection. This prevents a model from appearing secure simply by rejecting large numbers of benign requests.

Table 8. Comparative experimental results across authentication schemes.

Protocol	Latency (ms)	Energy (mJ)	Loss (%)	Benign success (%)	Attack rejection (%)
ECC-Baseline	34.53	1.097	2.14	99.10	91.20
LightECC	21.92	0.818	2.14	99.10	94.49
PSO-ECC	17.65	0.707	2.14	99.10	96.10
Proposed-QI-PSO-GWO	12.83	0.558	2.14	99.10	98.06

The packet-loss values in the benchmark are generated from network-load and attack-pressure factors and should be interpreted as comparative simulation indicators rather than radio measurements. A hardware/network experiment should measure packet loss directly under controlled RSSI, bandwidth, queue, and interference conditions.

Table 9 reports the results by traffic and attack type.

Table 9. Results by attack and traffic type.

Traffic class	Sessions	Authentication success / rejection	Latency (ms)	Energy (mJ)	Loss (%)
Benign	8626	99.10%	11.87	0.558	1.90
Credential	580	98.03%	14.93	0.557	1.90
DoS	727	98.09%	16.90	0.558	5.89
MITM	625	98.00%	14.84	0.553	1.91
Replay	715	98.04%	14.79	0.556	1.87
Spoofing	727	98.15%	14.80	0.557	1.86

For internet use being able to get in is the main thing that matters.. For bad traffic it is the opposite. When the system says yes to someone it is only good if that person is really supposed to be. If someone is trying to get in who should not be the system is working right if it says no. We should always do it this way when we test things in the future.

Critical versus non-critical devices

Table 10 compares performance for critical and non-critical device groups.

Table 10. Performance by device criticality group.

Device group	Sessions	Latency (ms)	Energy (mJ)	Mean authentication outcome (%)
Non-critical	8415	13.20	0.558	98.80
Critical	3585	11.96	0.557	98.82

The criticality feature is not meant to make security weaker for devices that're not critical. It actually helps the optimizer to put in effort to verify things where it really matters. For example when we are talking about infusion pumps, implantable devices and other safety critical equipment the minimum security rules should be much stronger in a world setup. This is because the criticality feature is about making sure that security is strong where it is needed the most, like, with infusion pumps and implantable devices.

Ablation design

A useful follow-up experiment is to remove one optimization component at a time: QI initialization only, PSO only, GWO only, and the full QI-PSO-GWO combination. The expected research question is whether the hybrid optimizer provides a statistically significant advantage over either single optimizer. The current benchmark establishes the end-to-end comparison but does not claim a formal ablation result.

Statistical and Sensitivity Analysis Plan

When we do research that's good enough for Scopus we should not just use one average value. We need to do the analysis times at least 20 times with different random starts for the optimization part. Each time we do this we use the group of benchmarks but we start the optimizer in a different way. We should write down the average and the standard deviation and also the range of values that we are 95 percent sure are correct for things like how it takes how much energy it uses how many packets get lost and how secure it is. We need to do this for all the metrics, like latency, energy, packet loss and security metrics so we can really understand what is going on with our Scopus-level research.

Sensitivity variables

Table 11 defines the low, nominal, and high values used in the sensitivity analysis.

Table 11. Sensitivity-analysis variables and ranges.

Variable	Low	Nominal	High
Network load	0.1	0.5	0.9
Battery level	20%	55%	90%
RSSI	-90 dBm	-65 dBm	-40 dBm
Attack rate	10%	28%	50%
Optimization population	10	30	60
Iterations	20	50	100
Critical-device fraction	10%	30%	50%

The tool that makes decisions needs to be tested hard when there are a lot of attacks and when the internet connection is not good. A strong plan should check carefully for sessions that are risky but not slow down real important traffic. Looking closely at how things change can show if the choices made for weights are steady or if the system relies too much on one way of setting things up.

Recommended statistical tests

- We use the paired Wilcoxon signed-rank test to compare differences, in each session when the numbers do not follow a pattern.
- When the numbers look normal we use the paired t-test to compare the sessions.
- To see how big the difference is we use Cliff's delta or Cohen's d.
- We also use bootstrap confidence intervals to see how well we can stop attacks and let good things go through.
- If we are comparing than two baseline sessions we need to make sure we are not getting false results so we use multiple-comparison correction.

The thing we are counting should be each authentication session or a group of devices and sessions not each cryptographic operation. This way we do not make it look like we have data than we really do..

Reproducibility checklist

- Share the dataset generator and the fixed random seed.
- Share the parameter files and the optimizer settings.
- Share the baseline implementations or the exact operation counts.
- Note the CPU model, RAM, operating system, Python version and cryptographic library.
- Mention the failed runs and the excluded sessions.
- Keep simulation results separate, from hardware measurements.

Discussion

The main insight from the benchmark is that optimization is best focused on policy selection of cryptographic primitives. The primitives should be. Well-audited building blocks, while the policy selection determines what and how much is being verified inside the security envelope. This has consequences for the reliability of any given optimizer because the chance of choosing an option that makes the system unsafe is lower.

The latency result confirms the design hypothesis. By moving the operations to the gateway or server the device only does a small constant-time online exchange. The energy result comes from the design choice. The question is how useful this optimization is in situations, which depends a lot on the specific use case. For some devices the radio's wake-up period and packet retransmission caused by link errors can be bigger than the encryption. Other factors like hardware acceleration the type of curve used radio features and network congestion can also be more important, than the cryptographic cost.

The security result shows that the second idea is correct and that using a made up threat is a choice. The system is very good at stopping attacks because it always checks if the information is new and it uses keys to make sure the messages are real.. We should not say that the system is safe from real attacks, without being careful. To really know if the system is safe we need to do a lot of tests. We need to check the system with tools try to break in and see how it works with real hardware. The security of the system and the security result are important to consider when talking about the systems security.

Comparison with recent literature

Recent work shows us a few ways to design things that work well together. The Selective-ECC authentication method (Julaihi et al., 2026) is useful because it only uses ECC when someone is signing up or during steps. Some authentication schemes (Amintoosi et al., 2022; Kim et al., 2023; Salim et al., 2024; Merchant et al., 2025) are really lightweight which helps reduce the cost of devices that're online. The PUF and authenticated-encryption designs (Aldosary and Tanveer, 2024) make a connection between the hardware and the device. We also have blockchain and edge approaches (Lau et al., 2023; Lin et al., 2024; Khan et al., 2025) that improve the way we trust each other and keep track of things. What the authentication framework I am talking about does is add a layer on top of all these methods so that the authentication policy can change when the resources and threats change. The authentication framework is important because it helps the authentication policy respond to changing resource and threat conditions. The authentication policy is what needs to be able to change when the resources and threats change and the authentication framework makes this possible by placing an optimization layer, above these mechanisms.

Practical deployment considerations

- Use secure boot and protected key storage for medical devices.
- Protect the gateway because it becomes a high-value enforcement point.
- Implement credential revocation and device de-registration.
- Use certificate/key rotation appropriate to device lifetime.
- Log authentication decisions without storing unnecessary patient data.
- Map security controls to applicable medical-device cybersecurity regulations and organizational policies.

Implementation Roadmap

Phase 1—Protocol implementation. Implement registration, gateway freshness screening, HMAC challenge-response, KDF session-key derivation, and revocation. Validate message syntax and state transitions.

Phase 2—Optimizer implementation. Implement QI initialization, PSO exploration, GWO refinement, objective calculation, feasibility constraints, and policy selection. Unit-test the optimizer independently from cryptographic code.

Phase 3—Network simulation. Integrate the protocol into ns-3 or another network simulator. Vary latency, bandwidth, packet loss, RSSI, node density, and attack intensity.

Phase 4—Formal verification. Model the authentication sequence in ProVerif/Tamarin/Scyther. Verify secrecy of the session key and authentication correspondence properties.

Phase 5—Hardware validation. Deploy a prototype on a Raspberry Pi-class gateway and representative constrained microcontroller nodes. Measure CPU time, memory, radio energy, packet retransmissions, and authentication latency using a power analyzer

Phase 6—Realistic security evaluation. Conduct replay, MITM, credential-stuffing, rogue-device, malformed-message, and request-flooding tests in an isolated laboratory environment

measurable outputs

Table 12 summarizes the measurable outputs and targets for the next phase.

Table 12. Measurable outputs and targets for the next phase

Output	Primary metric	Target for next phase
Authentication speed	End-to-end latency	<20 ms median
Device cost	Energy/session	Lower than ECC-heavy baseline
Security	Attack rejection	≥98% in benchmark; formal properties verified
Availability	Benign success	≥99%
Scalability	Sessions/second	Measured under 100–2000 devices
Memory	RAM/flash footprint	Reported for each device platform
Reproducibility	Independent rerun	Same trend across ≥20 seeds

5. Conclusion

This paper introduced an authentication system for devices in IoMT that uses optimization. The system deals with the problem of keeping different types of devices safe especially when they have limited resources. It doesn't make the authentication process too heavy for all devices. ECC is used to create a link, between the device and its identity when it is first set up. For the authentication it uses simple checks, HMAC or hash functions and keys made with KDF. A QI-PSO-GWO optimizer picks the authentication method. It looks at how important the device's how much battery is left how much network traffic there is, how fresh the data is and how risky the situation is.

A reproducible 12,000-session synthetic benchmark was created to compare ECC-Baseline, LightECC, PSO-ECC, and the proposed QI-PSO-GWO design. Under the stated simulation assumptions, the proposed framework achieved the lowest mean latency and energy cost and the highest attack rejection among the compared models. The findings support the hypothesis that adaptive policy selection can improve the security-performance balance.

The contribution should still be seen as a result from a simulation phase of the research. The next step is to check the protocol using public IoMT security data and limited hardware do checking, do statistical testing with many different starting points and see how strong it is, in real network and attack situations. The important next step is to test with real medical-device communication systems connected to hardware.

References

- [1] Guo, P., Xu, S. and Liang, W. (2025). A cloud-assisted anonymous and privacy-preserving authentication scheme for Internet of Medical Things. <https://doi.org/10.1016/j.cose.2025.104614>
- [2] Alam, I. and Kumar, M. (2023). A novel authentication protocol to ensure confidentiality among the Internet of Medical Things in COVID-19 and future pandemic scenario. <https://doi.org/10.1016/j.iot.2023.100797>
- [3] Alsaeed, N. and Nadeem, F. (2022). Authentication in the Internet of Medical Things: Taxonomy, Review, and Open Issues. <https://doi.org/10.3390/app12157487>
- [4] Alzahrani, F.A., Ahmad, M. and Ansari, M.T.J. (2022). Towards Design and Development of Security Assessment Framework for Internet of Medical Things. <https://doi.org/10.3390/app12168148>
- [5] Amintoosi, H., Nikooghadam, M., Shojafar, M., Kumari, S. and Alazab, M. (2022). Slight: A lightweight authentication scheme for smart healthcare services. <https://doi.org/10.1016/j.compeleceng.2022.107803>
- [6] Ghadi, Y.Y. et al. (2024). The role of blockchain to secure internet of medical things. <https://doi.org/10.1038/s41598-024-68529-x>
- [7] Julaihi, A.A. et al. (2026). SELAM: selective ECC-based lightweight authentication for the internet of medical things. <https://doi.org/10.1038/s41598-026-53641-x>
- [8] Khan, A.A., Laghari, A.A., Alroobaea, R. et al. (2025). A lightweight scalable hybrid authentication framework for Internet of Medical Things (IoMT) using blockchain Hyperledger consortium network with edge computing. <https://doi.org/10.1038/s41598-025-05130-w>

- [9] Kim, K., Ryu, J., Lee, Y. and Won, D. (2023). An Improved Lightweight User Authentication Scheme for the Internet of Medical Things. <https://doi.org/10.3390/s23031122>
- [10] Kumar, S., Abhishek, K. and Selvarajan, S. (2025). A lightweight and secure authentication and privacy protection scheme for internet of medical things. <https://doi.org/10.1038/s41598-025-05910-4>
- [11] Lau, C.H., Yeung, K.H., Yan, F. and Chan, S. (2023). Blockchain-based authentication and secure communication in IoT networks. <https://doi.org/10.1002/spy2.319>
- [12] Merchant, A., Panchami, V., Justine, S. and Eswaran, S. (2025). An efficient lightweight authentication scheme for smart healthcare in IoMT applications. <https://doi.org/10.1007/s10791-025-09779-9>
- [13] Mookherji, S., Vanga, O., Prasath, R. and Das, A.K. (2024). A secure authentication protocol for remote patient monitoring in an internet-of-medical-things environment. <https://doi.org/10.1002/spy2.428>
- [14] Aldosary, A. and Tanveer, M. (2024). PAAF-SHS: PUF and authenticated encryption based authentication framework for the IoT enabled smart healthcare system. <https://doi.org/10.1016/j.iot.2024.101159>
- [15] Praveen, R. and Pabitha, P. (2023). A secure lightweight fuzzy embedder based user authentication scheme for internet of medical things applications. <https://doi.org/10.3233/JIFS-223617>
- [16] Rana, A., Chakraborty, C., Sharma, S., Dhawan, S., Pani, S.K. and Ashraf, I. (2022). Internet of Medical Things-Based Secure and Energy-Efficient Framework for Health Care. <https://doi.org/10.1089/big.2021.0202>
- [17] Rexline, H.D. and X, A. (2026). A comprehensive survey on security and privacy in IoMT-driven smart healthcare: security attacks, mitigation measures, and future research direction. <https://doi.org/10.1007/s44443-026-00586-w>
- [18] Salim, M.M., Yang, L.T. and Park, J.H. (2024). Lightweight Authentication Scheme for IoT Based E-Healthcare Service Communication. <https://doi.org/10.1109/JBHI.2023.3338851>
- [19] Lin, Q., Li, X., Cai, K., Prakash, M. and Paulraj, D. (2024). Secure Internet of Medical Things (IoMT) based on ECMQV-MAC authentication protocol and EKMC-SCP blockchain networking. <https://doi.org/10.1016/j.ins.2023.119783>
- [20] Rasool, R.U., Ahmad, H.F., Rafique, W., Qayyum, A. and Qadir, J. (2022). Security and privacy of internet of medical things: A contemporary review in the age of surveillance, botnets, and adversarial ML. <https://doi.org/10.1016/j.jnca.2022.103332>
- [21] Su, X. and Xu, Y. (2024). Secure and Lightweight Cluster-Based User Authentication Protocol for IoMT Deployment. <https://doi.org/10.3390/s24227119>
- [22] Tamta, P. and Abhisheka, B. (2026). Securing the Internet of Medical Things (IoMT): A Comprehensive Taxonomy, Quantitative Benchmark, and Future Roadmap of Authentication Protocols. <https://doi.org/10.1111/exsy.70350>
- [23] Xu, F., Liu, S. and Yang, X. (2023). An efficient privacy-preserving authentication scheme with enhanced security for IoMT applications. <https://doi.org/10.1016/j.comcom.2023.06.012>
- [24] Zahednejad, B., Huang, T., Kosari, S. and Ren, X. (2023). A Lightweight, Secure Big Data-Based Authentication and Key-Agreement Scheme for IoT with Revocability. <https://doi.org/10.1155/2023/9731239>