International Journal of Safety and Security Engineering
Vol. 14, No. 3., June, 2024, pp. 807-813
Journal homepage: http://iieta.org/journals/ijsse

# An Efficient Lightweight Authentication and Access Control for IoT Edge Devices

Imane Zerraza*, Zianou Ahmed Seghir, Mounir Hemam

ICOSI Laboratory, Computer Science Department, Abbes Laghrour University, Khenchela 4000, Algeria

Corresponding Author Email: zerraza.imane@univ-khenchela.dz



https://doi.org/10.18280/ijsse.140313

**Received:** 2 January 2024
**Revised:** 9 May 2024
**Accepted:** 24 May 2024
**Available online:** 24 June 2024

***Keywords:***
*Internet of Things (IoT), authentication protocol, access control, Chacha20, Higher order attribute - Based Access Control (HoBAC)*

**ABSTRACT**

The emergence of Internet of Things (IoT) technology brings tremendous benefits to people's lives and work. However, the integration of IoT devices into diverse systems has underscored the pressing issue of security. Safeguarding data confidentiality in IoT systems necessitates the implementation of strong security measures, including authentication, encryption and access control mechanisms. When effectively employed, these measures pave the way for the development of an efficient and secure IoT system, offering substantial benefits to end-users. This paper introduces a novel authentication and access control solution tailored for IoT edge devices. Our suggested approach is applicable to the edge network comprising numerous nodes, enabling the transmission of extensive data within constrained bandwidth based on lightweight symmetric cryptography since it uses Chacha20 algorithm to establishing session key, and maintaining the protocol of identity management and access control more precisely through HoBAC and blockchain technology. We have successfully analyzed the protocol correctness using the scyther tool. The results demonstrate the superiority of the proposed protocol compared to alternative approaches, particularly in terms of communication and time costs.

## 1. INTRODUCTION

The growing number of Internet of Things (IoT) devices has the potential to collect and share massive amounts of data. To ensure that these data are securely collected and handled, decentralization is a key strategy. Decentralization involves the use of multiple local computing devices or cloudlets located close to the data source, which helps to reduce latency and improve data security. Additionally, identity management solutions such as decentralized identity management can be used to ensure secure authentication and authorization for IoT devices [1]. Each connected device does need to have a unique identification and addressing scheme in order to securely participate in the IoT. This allows it to be accurately identified and accessed, and also helps to ensure the security of the data that is exchanged between the devices in the network. However, due to the wide use of wireless sensor networks (WSN), valuable information is transmitted across networks in an unsafe manner, thus making it crucial to have secure data transmission protocols in place to protect the sensitive data. Given the limited power, memory, and processing capabilities of Wireless Sensor Networks (WSNs), these systems are highly susceptible to security attacks. Therefore, it is crucial to implement robust security protocols to safeguard the integrity and confidentiality of the data [2]. It can be difficult to implement traditional security measures on resource-constrained IoT devices, as these security mechanisms were originally designed for desktop computers. Therefore, it is important to explore other security protocols that are specifically suited for the resource-constrained environment of IoT devices and networks. One such protocol that is designed for the IoT is called 6LoWPAN, which enables secure data transmission of IPv6 packets over the IEEE 802.15.4 wireless network. In order to secure the communication of IoT devices against malicious attacks, the deployment of security protocols such as 6LoWPAN can be highly effective. One of the most efficient methods of ensuring the security of IoT networks is through encryption. To provide an additional layer of security to IoT networks, lightweight cryptography can be implemented due to its sophisticated nature. Additionally, hardware security plays a crucial role in protecting both devices and algorithms in IoT networks [3]. Several techniques have been proposed to ensure hardware security, such as hiding or masking. These methods help to protect data and devices from malicious attacks, while also allowing encrypted data to be effectively transferred between network entities [4]. There are many security and privacy measures that can be implemented at the physical level, these measures cannot guarantee immunity against side channel and hardware attacks. To provide additional protection, it is important to combine these physical security measures with other technologies such as firewalls, intrusion detection systems, and authentication protocols to prevent malicious actors from gaining access to the system. Additionally, implementing secure encryption algorithms and key management systems can help protect data from side-channel and hardware attacks [5]. Secure and reliable identity is essential for identifying a user accurately and providing services, as well as allowing

access to resources and features on a secure network. To provide secure identity management, organizations utilize strong authentication protocols such as two-factor authentication or biometrics to verify users. Additionally, secure encryption algorithms and digital signature systems can help ensure that the identity is kept secure from malicious actors. Decentralized identity management systems are gaining popularity, as they provide more transparency, improved communications, and cost savings. This approach removes the need for a central governing authority, and allows individuals and devices in the IoT to manage their own personally identifiable information securely and reliably. To ensure the security of data, traditional access control technologies are not suitable for use in IoT systems due to centralized control and complex access management. Therefore, it is important to use technologies such as blockchain, distributed ledgers, and decentralized trust networks that are specifically designed for IoT systems to ensure secure access control [6]. The contribution of this paper is listed as follows:

1) We propose a lightweight authentication protocol for IOT devices in an edge environment. Our design uses a symmetric cryptography to establish session key using Chacha20.

2) We restrict the privilege of users by using HoBAC and blockchain technology.

3) We enhance the security against different types of attacks, such as side channel attack and sybil attack.

4) We evaluate the performance of our protocol.

The rest of this paper is organized into several sections. Section 2 of the paper will discuss various security and privacy measures for authentication protocols. Section 3 of the paper will discuss an authentication and Higher order attribute - Based Access Control (HoBAC) policy solution for the Internet of Things. Section 4 of the paper will discuss the implementation of the proposed solution. Lastly, section 5 will conclude the paper.

## 2. RELATED WORK

In previous research [7], various authentication schemes have been reviewed to enhance understanding of this security mechanism. Additionally, this section will discuss several proposed works aimed at bolstering IoT security through the implementation of authentication protocols represented in Table 1.

In the study of Salami et al. [8], a lightweight encryption scheme for smart homes based on Stateful Identity-Based Encryption (IBE) is proposed, in which the public keys are merely identity strings without the need for a digital certificate. This scheme uses identities of the sender and receiver to encrypt and decrypt data in order to provide secure communication between devices. Additionally, because it does not require the use of digital certificates, it is more lightweight and easier to implement than other solutions. This makes it suitable for use in smart home networks and other connected devices. Phong, Matsuka, and Ogata (PMO) Stateful Identity-Based Encryption (IBE) scheme is a combination of IBE and Stateful Diffie-Hellman (DH) encryption scheme. This scheme divides the encryption process into two sub-algorithms: key encryption and data encryption, so that only data ciphers are transmitted multiple times without attaching the key ciphertext. The evaluation results showed that the proposed scheme is secure against plaintext attacks, making it a suitable choice for secure communication in IoT networks.

Santoso and Vun [9] presented a secure authentication protocol employing ECC for IOT based smart homes. The suggested system utilised a gateway centric AllJoyn framework, offering an improved authentication interface for Android devices. The scheme provides secure mutual authentication between entities with fewer message overheads.

The work of Ola et al. [10] introduces a Software Defined Network (SDN) identity-based authentication scheme for IoT. This scheme uses a shared identity based on virtual IPv6 through the SDN controller to translate the different identity formats used by various communication protocols. Additionally, the gateways are responsible for authenticating devices while the gateways themselves are authenticated by the central controller. This scheme is designed to resist common attack types such as replay, man-in-the-middle, and masquerade attacks, making it a suitable choice for secure communication in IoT systems.

In the study of Ye et al. [11], an efficient authentication technique based on Elliptic Curve Cryptography (ECC) to secure the perception layer of the Internet of Things (IoT). However, it does not address the attribute-based access control policies among devices. This could be an interesting problem to look into further, as developing a secure, robust access control policy for IoT devices is necessary for reliable operation.

Witkovski et al. [12] proposes a solution for authenticating IoT devices based on session keys and symmetric cryptography. This could be beneficial in a smart home scenario, where the gateway can integrate the traditional internet and IoT to maintain the electronic devices securely. This type of authentication scheme could help ensure a high level of security for devices in a smart home.

**Table 1.** Summary of relevant authentication protocols

| Protocols | Authentication Scheme | Advantages | Limitations |
|---|---|---|---|
| [8] | Identity based encryption | Encyption process contains two sub-algorithm. | Lacks extensive.<br>Cannot resist side channel attack. |
| [9] | Smart home based authentication | Provide mutual authentication. | Cannot resist Sybil attack. |
| [10] | Identity based authentication | Gateways authenticating devices.<br>Use shared identity. | Lacks access control policies.<br>Cannot resist Sybil attack |
| [11] | ECC based authentication | Provide mutual authentication.<br>Large scale. | Lacks access control policies. |
| [12] | IdM and key based authentication | Use symmetric cryptography.<br>Low power. | Cannot resist side channel attack. |
| [13] | Chacha20 poly 1305 authenticated encryption | Use symmetric cryptography.<br>Low power. | Cannot resist side channel attack. |
| [14] | Chacha20 data encryption | Use symmetric cryptography.<br>Low power. | Cannot resist side channel attack. |

In the study of Santis et al. [13], the authors evaluated the suitability of Chacha20 - Poly1305 authenticated encryption with associated data (AEAD) scheme for ARM Cortex-$M_4$ processors. The study showed that it is faster compared to hardware-accelerated AES-128 in GCM, EAX, and CCM modes. This could be beneficial for applications that require high-speed and lightweight encryption, such as IoT applications.

Reza et al. [14] Suggested an approach that combines the Chacha20 data encryption method, chaos-based key generation and public key-based authentication. Based on mathematical analysis, the proposed scheme demonstrates compatibility with Smart Grids (SGs), offering advantages such as reduced power consumption, faster processing times, and increased throughput. These attributes render it both quicker and more resource-efficient. This lightweight security scheme prevents replay attacks or any other timing attacks, making it a secure solution for IoT networks.

## 3. AUTHENTICATION AND HOBAC POLICY

An architecture and protocol consisting of three phases to secure IoT networks is proposed. The registration phase is used to register edge nodes with a trusted server, the authentication phase is used to verify the identity of nodes, and the establish session key phase is used to securely exchange keys between nodes. This proposed protocol could be beneficial for ensuring a secure, reliable connectivity between IoT devices.

### 3.1 Architecture

In this work, an infrastructure based on edge computing (see Figure 1) [15] and blockchain technology to secure IoT networks is proposed. Edge device (acting as a WSN [11]) is used to collect data and an edge server is used to join the infrastructure, with a set of base stations assigned to it.

The proposed protocol uses a blockchain for identity creation and High Order Attribute-based access control policies to restrict user privileges. Additionally, it uses the symmetric Chacha20 algorithm for encryption. This could be an effective, secure solution for protecting data in IoT networks.

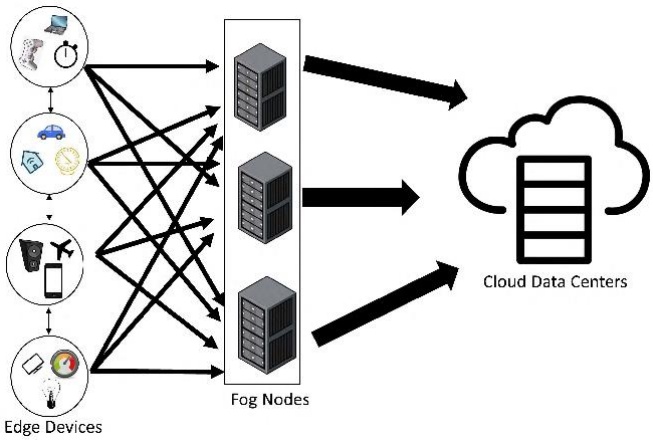


**Figure 1.** Edge computing architecture

Before disclosing our proposed protocol, it is essential to specify the adversary model utilized. An adversary Adv with polynomial time capabilities exercises total command over the insecure network traffic with the intention of undermining the security of the proposed scheme. Adv has the ability to oversee partial/entire messages conveyed via an insecure channel, such as intercepting, altering, and erasing the transmitted message. Adv has the capability to retrieve the security information embedded on a smart card utilizing a power analysis method. Adv could endeavor to gather confidential data, such as passwords utilizing side-channel attacks. The objective of Adv is to accomplish one of the following: Calculate the session key following a successful execution of the authentication scheme. Derive the server's long-term secret key. Make the server erroneously approve an authentication scheme when not engaged with a genuine entity.

### 3.2 Authentication protocol

Under the same intensity of security compared to other cryptography, Chacha20 has many advantages such as its adaptation to resource-constrained devices [16]. Therefore, a method of mutual authentication using Chacha20 can be used to establish a secure secret key in an IoT network. This method helps ensure a secure connection between a group of devices and allows data to be shared in a safe and secure manner. The notations used in the rest of the paper are show in Table 2.

**Table 2.** Notations and abbreviations

| Notation | Descriptions |
|---|---|
| $ID_{user}$, $ID_{Node}$ | The identity of user/ node |
| $PK_{user}$, $PK_{Node}$ | The public key of user/ node |
| $H_1$, $H_2$ | Hash function |
| Com | The commitment scheme |
| $\gamma$ | The group commitment |
| $q_i$ | Element of the unique commitment vector |
| $v_i$ | The opening |
| n | The number of k (n= \|k\|) |
| k | Preshared secret key |
| IDK | The node partiel key |
| $\oplus$ | XOR operation |
| $\|\|$ | Concatenation operation |
| (E, D) | Chacha20 encryption scheme |
| mk | The secret key of base station |
| Params | A set of public parameters genereted by BS |

3.2.1 Edge node registration phase

Each device (node or user) can get its ID using blockchain technology combined with threshold public key signatures [17]. This allows devices to securely exchange data and also creates a secure environment for executing smart contracts. Additionally, blockchain technology helps ensure that the devices remain anonymous while still utilizing the full benefits of distributed ledger technology.

After receiving the smart contract, the edge server can store it in a secure database. This ensures that the data stored is safe from any malicious activity or unauthorized access. Additionally, the edge server can use various security measures such as encryption, authentication, and authorization to protect the data stored in the database.

3.2.2 Authentication phase

The user sends to the node ($N_1$, H (K, $N_1$)).

The node receiving the message and send ($N_2$, $sig_{PKuser}$ ($N_2$, $N_1$, $ID_{Node}$), H (K, $N_2$)).

Once the user receiving the message, it sends ($sig_{PKNode}$ ($N_1$, $N_2$, $ID_{user}$)). Figure 2 presents this process.

K: pre-shared secret between two entities. This preshared secret (K) is an important security tool when it comes to secure communication between two entities. This secret can be used as a password or key for authentication and encryption purposes, ensuring that the data being transmitted is kept safe from any malicious third parties. Additionally, the pre-shared secret can be used in conjunction with other security measures such as digital certificates and access control mechanisms to further enhance security.

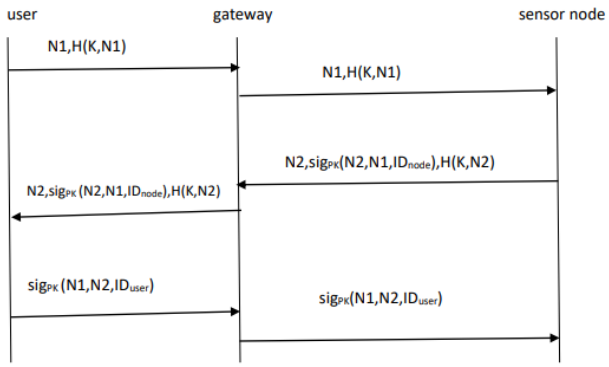


**Figure 2.** Authentication phase

N1, N2: random number. The pseudorandom N1 and N2 are numbers that are generated by the two entities in secure communication. These numbers are used as additional security measures to ensure secure communication between the two entities. They are typically generated at the beginning of the communication process and used as a part of authentication and/or encryption processes.

H: hash function. A hash function, or H, is a cryptographic tool used to generate a unique digital representation of data. A hash function takes in data and returns an output that is of fixed length and cannot be reversed. This output can then be used for various purposes, such as data integrity verification and secure communication. Hashing functions are typically used in combination with other security measures, such as encryption and access control mechanisms.

3.2.3 Establish session key phase

Setup. The base station (BS), performing as the key management servers in ATSE [17], generates $G_1$ and $G_2$, both of which have the same prime order q, and selects an admissible pairing e: $G_1 * G_1 \rightarrow G_2$, where g is a generator of $G_1$.

It then picks ∈ Zq at random and computes y = g s. The BS additionally chooses a symmetric encryption scheme (E, D), with E representing encryption and D representing decryption operations.

The BS also opts for a pair of hash functions. $H_1$: {0, 1} * → G1 and $H_2$:$G_2$ → 0, 1 n for some positive integer n, which is the size of the key.

The BS sets secret master key mk = s and a set of public parameters params = ($G_1$, $G_2$, e, g, y, (E, D), $H_1$, $H_2$). The BS shares params with all connected entities.

Setup. Upon receiving an identity $ID_{Node}$ the BS check whether the requester's identity is indeed $ID_{Node}$ and return IDK = $e(H_1(ID_{Node}))s$ as partiel key.

KeyEncrypt. The sender performs the following:

• pick a symmetric key K.

• ifst=null, undertake the subsequent tasks:

• pick r ∈ zq uniformly at random.

• set st = (r, $g^r$, $e(H_1(ID_{Node}), y)^r$).

• else do the following:

• parse st as (r, $g^r$, $e(H_1(ID_{Node}), y)^r$).

• pick a sample random value $v_i$.

• compute the commitment γ and $q_i$.

• for each i ∈ [n] sender executes:

– compute the message specific whole key $wk_i = e(H_1(ID_{Node}), H_2(com(k_i, v_i)))^s$.

– generate cki = ($ID_{Node}$, $H_2$(com ($k_i$, $v_i$)), γ, $q_i$, $e_i$) where $e_i$ = PRG ($wk_i$) ⊕ ($k_i$ ||$v_i$).

– Output ck = ($ck_1$, $ck_2$, $ck_N$) as "key ciphertext".

DataEncrypt. Find the symmetric key K, then carry out the following tasks:

• compute $c_m = E_k(m)$.

• return $c_m$ as "data ciphertext".

DKeyDecrypt. After receiving the key ciphertext ck, the receiver proceeds to execute the following actions with the $ID_{Node}$:

• compute wk = ($H_1$(IDK), $H_2$(com($k_i$ , $v_i$))).

• decrypt e as (k||v) = prg(wk) ⊕ e.

• verify the commitment γ with opening $v_i$; if it returns 0 then output ⊥, else output k.

DataDecrypt. After receiving data ciphertext $c_m$, the receiver locates k and then employs k to decrypt the symmetric key, proceeding with the following actions:

• compute $m = D_k(c_m)$.

• return m.

### 3.3 Higher order attribute based access control model

Ensuring security in dynamic IoT environments is a prominent and challenging task. Access control is one of the essential aspects of data security, as it helps to control access to information based on pre-defined access policies. HoBAC [18] is a new access control model that is a generalization of ABAC, and helps to create flexible access control policies for both IoT and non-IOT systems, by building hierarchies of entities based on attributes. The HoBAC model also supports fine composition and aggregation operations, allowing for highly adaptable access policies in a variety of real-world scenarios. The HoBAC model consists of three main components: entities, subjects, objects and contexts. Entities are the combination of the three basic concepts of ABAC (subject, object and context), and represent the entities that request access to an object in order to perform an operation. Subjects are the entities that request access to an object, while objects are the resources requested by the subject. Contexts are the operational contexts in which the access requests occur. Together, these components form the basis of the HoBAC policy. Attributes in the HoBAC model are defined by an ID and a value, where the ID is a unique identifier that can be used to identify the attribute itself, and the value is a value of some type. Attribute IDs are assigned to each entity in the system, and can then be used to set access control policies based on the values associated with the attributes. Access control rule or access policies in HoBAC are represented by rules that determine whether an access request should be allowed or denied. Access decisions are based on an attribute-based strategy that evaluates the values of attributes from subjects, objects and contexts, and checks a set of rules specified in terms of these attributes. The access policy can be customized using the attribute-based system to create a powerful, yet flexible access control system for IoT and nonIOT systems.

Aggregation operations allow the federation of attributes from entities, allowing a single object (subject or context) to be created with a high level of abstraction using one or more attributes. This makes it possible to create access control policies that are tailored to specific requirements, and helps to ensure that only authorized users have access to the required resources. The access control policy process in HoBAC allows multiple subjects or objects to be combined using aggregation operations, which creates a single request for access or resource privileges rather than sending multiple requests. This abstraction process helps to provide an additional layer of security, as it prevents direct manipulation of low-level subjects and objects. Additionally, this process makes it easier to manage and enforce the access control policies in the system, as fewer requests and queries need to be handled, allowing for a more secure and reliable identity management system.

## 4. IMPLEMENTATION

The edge device was implemented using the Python framework PyFUNS (python framework for ubiquitous networked sensors (PyFUNS was designed for resource-constrained devices, providing an intuitive and user-friendly API for building IoT applications. It also supports ContikiOS, making it possible to easily deploy applications in a distributed environment and across multiple platforms.)) [19], while Chacha20 algorithm was used to encrypt the messages. The protocol scheme was implemented in Python, and executed on ContikiOS, which is an open source operating system designed specifically for the IoT. This platform was chosen due to its ability to support resource-constrained devices, as well as its low memory and processing overhead. This makes it an ideal platform for secure data transmission and authentication protocols in IoT networks. Additionally, ROSITA++ [20] was used to detect and eliminate higher order leakage, and to automatically protect a masked implementation of Chacha [21]. These measures help to ensure secure data transmission and protection from malicious actors, thus making it safer for sensitive information to be exchanged between network entities.

### 4.1 Security proof with scyther

Scythe [22] emerges as a potent and efficient tool for the thorough examination and identification of potential security breaches and vulnerabilities within security protocols. Its automated analysis capabilities enable comprehensive scrutiny of protocol behavior, effectively assessing its resilience against a spectrum of potential attacks. One of Scythe's standout features, Niagree, offers assurance to communicating parties regarding the secure transmission and correct sequencing of messages, bolstering confidence in the integrity of data exchanges. Furthermore, the inclusion of the Alive feature serves to validate protocol steps, ensuring proper authorization by involved parties and mitigating the risk of unauthorized access. Additionally, Scythe's Weakagree feature acts as a crucial defense mechanism against impersonation attacks, further fortifying the security posture of IoT networks. Through the integration of these advanced features, Scythe facilitates the establishment of a secure and reliable environment for data transmission within IoT networks, instilling trust and confidence in the integrity of communication channels.

### 4.2 Security analysis of the proposed protocol

Mutual authentication serves as a mechanism to verify that communication occurs between intended parties. Given that 'K' represents a long-term shared value utilized for mutual authentication, it implies that the two entities have previously exchanged this key. To ensure secure authentication, the shared key should be securely protected from malicious actors, and protected from unauthorized access.

Side channel attacks are a type of attack that can be used to examine cryptographic algorithms and exploit information related to power consumption, execution timing, and electromagnetic fields. To protect against these types of attacks, our proposed protocol utilizes the ROSITA tool to protect the masked implementation of Chacha, which eliminates over 99% of the leakage from cryptographic implementations. This makes it difficult for attackers to obtain any useful information from the encryption devices, thus helping to ensure secure data transmission in IoT networks.

In order to protect against a Sybil attack, our protocol makes use of secure authentication protocols such as two-factor authentication or biometrics to verify that users are legitimate. Additionally, implementing access control techniques such as Higher order attribute-based access control (HoBAC) and secure encryption algorithms can also help to mitigate the risk of a Sybil attack. Finally, using distributed trust networks and blockchain technology can help ensure that only authenticated users are able to access the network.

Secure against the reply attack: the use of commitment and random numbers in the proposed protocol helps to protect against reply attacks, as the attacker would not be able to send duplicate messages. The commitment ensures that every step is verified, while the random numbers create unique values for every session, making it difficult for an attacker to resend the same message. By implementing these measures, the system can help to ensure secure data transmission and protect against malicious actors.

Man-in-the-middle attack: This refers to a form of cyberattack in which the attacker covertly intercepts and forwards messages between two entities who are under the impression that they are communicating directly. Because it is impossible to know the pre-shared key between the entities. Additionally, it is arduous for an attacker to obtain useful information ($N_1$, $N_2$, $ID_{Node}$) from messages sent to user.

Secure against crypanalysis attacks: These attacks are focused on the ciphertext and they try to break the encryption by finding the encryption key to obtain the plaintext. To protect against this kind of attack, you can use a session key which is transmitted separately in a secure sub algorithm without attaching the data ciphertext.

User anonymity: Let's assume an attacker possesses a message denoted as $A_{IDnode} = (N_2, sig_{PKuser} (N_2,N_1, ID_{Node}), H(K,N_2))$. However, extracting $ID_{Node}$ is exceptionally challenging due to the necessity of the verification of the signature using the private key of user, a piece of information to the user. In addition, it is difficult to get the random number $N_1$, $N_2$ and K the pre-shared key. As a result, our scheme firmly establishes strong user anonymity.

User un-traceability: Let's consider an attacker A, who intercepts two messages, $A_{IDnode}$ and $A_{IDnode'}$, derived from an openly accessible channel with the aim of tracing the user. In the scenario that any element in both messages matches, A could deduce that the messages are sent by the same user. However, our scheme adeptly prevents user un-traceability

attacks since the values in $A_{IDnode}$ and $A_{IDnode}$' are formed using a random nonce, N1 and N2. Hence, these values are distinct for each session and persist as valid solely within that particular session.

## 5. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed protocol by performing comparisons with other protocols, such as those proposed by Shahidinejad et al. [23], Chen et al. [24], Chen et al. [25] and Kumer and Om [26], in terms of the computation time and communication cost.

The computation times of these protocols, including our proposed one, are illustrated in the Figure 3. From the bar chart analysis, it is evident that our protocol exhibits the shortest computation time compared to those in the studies of Shahidinejad et al. [23], Chen et al. [24], Chen et al. [25] and Kumer and Om [26], with about 13ms, 41ms, 24.77ms and 30.9166ms.

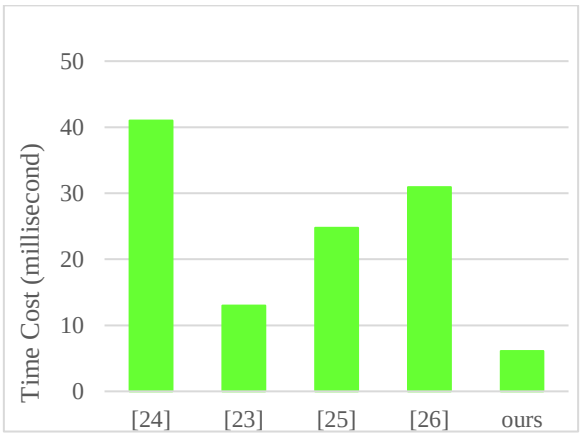


**Figure 3.** Time cost comparison

In the comparison of communication costs, as illustrated in Figure 4, our proposed protocol outperforms other related protocols. The communication cost for our scheme is 1552 bits, while the counterpart protocol by Shahidinejad et al. [23] and Chen et al. [24], Chen et al. [25], Kumerand Om [26], incurs a higher cost of 1954 bits, 3904 bits, 2496 bits and 4800 bits respectively. This disparity highlights the cost-effectiveness of our proposed scheme in communication expenditure, and it further emphasizes the comprehensive security features embedded in our solution, safeguarding against a spectrum of security attributes and potential threats.

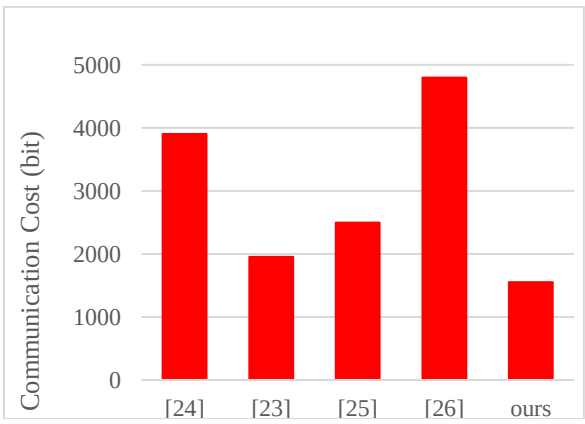


**Figure 4.** Communication cost comparison

## 6. CONCLUSION AND FUTURE WORKS

The surge in the number of connected devices has given rise to a multitude of security risks and challenges for IoT systems. The widespread distribution of IoT devices creates significant difficulties in enforcing security authentication and access control, which renders them vulnerable to various types of attacks by malicious adversaries. As such, it is imperative to develop robust security measures that can address the unique challenges posed by IoT networks. By continuously enhancing and updating these measures, we can help ensure that IoT systems remain secure and safeguard the privacy of their users. Our proposed authentication and access control scheme has been demonstrated to be highly effective in securing Wireless Sensor Networks (WSNs) in IoT systems. The utilization of an edge architecture has been instrumental in reducing network latency and improving response times, thereby enhancing the overall performance of the system. By using a symmetric algorithm to create a session key, we have ensured that the system is equipped with robust security mechanisms. Overall, the successful implementation of the proposed scheme in a WSN-based IoT system underscores the importance of developing efficient and secure authentication and access control protocols to address the unique security challenges posed by IoT networks. To confirm the security of our proposed protocol, we examined the protocol using the scyther tool to ensure its sturdiness against diverse attack modalities. We extensively compared our proposed protocol with alternative protocols employed in edge computing infrastructure. Our findings indicate that both the computation time and communication cost of our proposed protocol are notably lower than those of alternative protocols. Additionally, our proposed protocol excels in terms of security and maintains a lightweight design, positioning it as a favorable choice within the domain of edge computing.

In the future, research will involve additional experimentation, incorporating alternative architectures like fog computing. The investigation will also delve into encryption algorithms, including AES, and explore various symmetric and asymmetric algorithms.

## REFERENCES


[1] Zadeh, H.H., Soyata, T., Kantarci, B., Boukerche, A., Kaptan, C. (2018). Sensing, communication and security planes: A new challenge for a smart city system design, Computer Networks, 144: 163-200. https://doi.org/10.1016/j.comnet.2018.08.001

[2] Khattak, H.A., Shah, M.A., Khan, S., Ali, I., Imran. M. (2019). Perception layer security in Internet of Things. Future Generation Computer Systems, 100: 144-164. https://doi.org/10.1016/j.future.2019.04.038

[3] Hameed, A., Alomary, A. (2019). Security issues in IoT: A survey. In 2019 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), Sakhier, Bahrain, pp. 1-5. https://doi.org/10.1109/ 3ICT.2019.8910320

[4] Jungk, B., Petri, R., Stottinger, M. (2018). Efficient side-channel protections of ARX ciphers. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2018(3): 627-653. https://doi.org/10.13154/tches.v2018.i3.627-653

[5] Habibzadeh, H., Nussbaum, B.H., Anjomshoa, F., Kantarci, B., Soyata, T. (2019). A survey on cybersecurity, data privacy, and policy issues in cyber-physical system deployments in smart cities. Sustainable Cities and Society, 50: 101660. https://doi.org/10.1016/j.scs.2019.10166

[6] Zhu, X.Y., Badr, Y. (2018). A Survey on blockchain-based identity management systems for the Internet of Things. In 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Halifax, NS, Canada, pp. 1568-1573. https://doi.org/0.1109/Cybermatics_2018.2018.00263

[7] Bangare, P.S., Patil, K.P. (2022). Study and analysis of various authentication and authorization for IoT devices: A challenging overview. International Journal of Safety and Security Engineering, 12(2): 209-216. https://doi.org/10.18280/ijsse.120209

[8] Salami, S.A., Baek, J., Salah, K., Damiani, E. (2016). Lightweight encryption for smart home. In 2016 11th International Conference on Availability, Reliability and Security (ARES), Salzburg, Austria, pp. 382-388. https://doi.org/10.1109/ARES.2016.40

[9] Santoso, F.K., Vun, N.C.H. (2015). Securing IoT for smart home system. In 2015 International Symposium on Consumer Electronics (ISCE), Madrid, Spain, pp. 1-2. http://doi.org/10.1109/ISCE.2015.7177843

[10] Ola, S., Abdallah, S., Elhajj, I.H., Chehab, A., Kayssi, A. (2016). Identity-based authentication scheme for the Internet of Things. In 2016 IEEE Symposium on Computers and Communication (ISCC), Messina, Italy, pp. 1109-1111. http://doi.org/10.1109/ISCC.2016.7543884

[11] Ye, N., Zhu, Y., Wang, R.C., Malekian, R., Lin, Q.M. (2014). An efficient authentication and access control scheme for perception layer of Internet of Things. Applied Mathematics and Information Sciences, 8(4): 1617-1624. http://doi.org/10.12785/amis/080416

[12] Witkovski, A., Santin, A., Abreu, V., Marynowski, J. (2015). An IdM and key-based authentication method for providing single sign-on in IoT. In 2015 IEEE Global Communications Conference (GLOBECOM), San Diego, CA, USA, pp. 1-6. https://doi.org/10.1109/GLOCOM.2015.7417597

[13] Santis, F.D., Schauer, A., Sigl, G. (2017). Chacha20-Poly1305 authenticated encryption for high-speed embedded IoT applications. In Design, Automation and Test in Europe Conference and Exhibition (DATE), 2017, 2017, Lausanne, Switzerland, pp. 692-697. https://doi.org/10.23919/DATE.2017.7927078

[14] Reza, S.M.S., Ayob, A., Arifeen, M., Amin, N., Saad, M.H.M., Hussain, A. (2020). A lightweight security scheme for advanced metering infrastructures in smart grid. Bulletin of Electrical Engineering and Informatics, 9(2): 777-784. http://doi.org/10.11591/eei.v9i2.2086

[15] Hassija, V., Chamola, V., Saxena, V., Jain, D., Goyal, P., Sikdar, B. (2019). A survey on IoT security: Application areas, security threats, and solution architectures. IEEE Access, 7: 82721-82743. https://doi.org/10.1109/ACCESS.2019.2924045

[16] Bernstein, D. (2008). Chacha, a variant of Salsa20. SASC 2008 – the State of the Art in Stream Ciphers. https://cr.yp.to/chacha.html,2008.

[17] Christodorescu, M., Gaddam, S., Mukherjee, P., Sinha, R. (2021). Amortized threshold symmetric-key encryption. In Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security, pp. 2758–2779. https://doi.org/10.1145/3460120.3485256

[18] Aliane, L., Adda, M. (2019). HoBAC: Toward a higherorder attribute-based access control model. Procedia Computer Science, 155: 303-310. https://doi.org/10.1016/j.procs. 2019.08.044

[19] Bocchino, S., Fedor, S., Petracca, M. (2015). PyFUNS: A python framework for ubiquitous networked sensors. Wireless Sensor Networks, Springer, Cham, pp. 1-18. https://doi.org/10.1007/978-3-319-15582-1_1

[20] Shelton, M.A., Chmielewski, Ł., Samwel, N., Wagner, M., Batina, L., Yarom, Y. (2021). Rosita++: Automatic higher-order leakage elimination from cryptographic code. In CCS. Special Interest Group on Security, Audit and Control (SIGSAC) of the Association for Computing Machinery (ACM), pp. 685-699. http://doi.org/10.1145/3460120.3485380

[21] Shelton, M.A., Samwel, N., Batina, L., Regazzoni, F., Wagner, M., Yarom, Y. (2021). Rosita: Towards automatic elimination of power-analysis leakage in ciphers. In 28th Annual Network and Distributed System Security Symposium, pp. 21-25. http://doi.org/10.14722/ndss.2021.23137

[22] https://people.cispa.io/cas.cremers/scyther/, accessed on Mar. 16, 2024.

[23] Shahidinejad, A., Ghobaei-Arani, M., Souri, A., Shojafar, M., Kumari, S. (2021). Light-edge: A lightweight authentication protocol for IoT devices in an edge-cloud environment. IEEE Consumer Electronics Magazine, 11(2): 57-63. http://doi.org/10.1109/MCE.2021.3053543

[24] Chen, C.M., Chen, L.L., Huang, Y.Y., Kumar, S., Wu, J.M.T. (2021). Lightweight authentication protocol in edge-based smart grid environment. EURASIP Journal on Wireless Communications and Networking, 2021: 68. https://doi.org/10.1186/s13638-021-01930-6

[25] Chen, C.M., Chen, Z.T., Kumari, S., Lin, M.C. (2022). LAP-IoHT: A lightweight authentication protocol for the internet of health things. Sensors, 22(14): 5401. https://doi.org/10.3390/s22145401

[26] Kumar, P., Om, H. (2024). Multi-TA model-based conditional privacy-preserving authentication protocol for fog-enabled VANET. Vehicular Communications, 47: 100785. https://doi.org/10.1016/j.vehcom.2024.10078