Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.049017

ARTICLE

A Hybrid and Lightweight Device-to-Server Authentication Technique for the Internet of Things

Shaha Al-Otaibi[1], Rahim Khan[2,*], Hashim Ali[2], Aftab Ahmed Khan[2], Amir Saeed[3] and Jehad Ali[4,*]

[1]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P. O. Box 84428, Riyadh, 11671, Saudi Arabia

[2]Department of Computer Science, Abdul Wali Khan University Mardan, Mardan, 23200, Pakistan

[3]Department of Computer Science and IT, UET Peshawar, Jalozai Campus Peshawar, Peshawar, 24240, Pakistan

[4]Department of AI Convergence Network, Ajou University, Suwon, 16499, South Korea

*Corresponding Authors: Rahim Khan. Email: rahimkhan@awkum.edu.pk; Jehad Ali. Email: jehadali@ajou.ac.kr

Received: 25 December 2023 Accepted: 29 January 2024 Published: 26 March 2024

ABSTRACT

The Internet of Things (IoT) is a smart networking infrastructure of physical devices, i.e., things, that are embedded with sensors, actuators, software, and other technologies, to connect and share data with the respective server module. Although IoTs are cornerstones in different application domains, the device's authenticity, i.e., of server(s) and ordinary devices, is the most crucial issue and must be resolved on a priority basis. Therefore, various field-proven methodologies were presented to streamline the verification process of the communicating devices; however, location-aware authentication has not been reported as per our knowledge, which is a crucial metric, especially in scenarios where devices are mobile. This paper presents a lightweight and location-aware device-to-server authentication technique where the device's membership with the nearest server is subjected to its location information along with other measures. Initially, Media Access Control (MAC) address and Advance Encryption Scheme (AES) along with a secret shared key, i.e., λ_i of 128 bits, have been utilized by Trusted Authority (TA) to generate MaskIDs, which are used instead of the original ID, for every device, i.e., server and member, and are shared in the offline phase. Secondly, TA shares a list of authentic devices, i.e., server S_j and members C_i, with every device in the IoT for the onward verification process, which is required to be executed before the initialization of the actual communication process. Additionally, every device should be located such that it lies within the coverage area of a server, and this location information is used in the authentication process. A thorough analytical analysis was carried out to check the susceptibility of the proposed and existing authentication approaches against well-known intruder attacks, i.e., man-in-the-middle, masquerading, device, and server impersonations, etc., especially in the IoT domain. Moreover, proposed authentication and existing state-of-the-art approaches have been simulated in the real environment of IoT to verify their performance, particularly in terms of various evaluation metrics, i.e., processing, communication, and storage overheads. These results have verified the superiority of the proposed scheme against existing state-of-the-art approaches, preferably in terms of communication, storage, and processing costs.

KEYWORDS

Internet of things; authenticity; security; location; communication

1 Introduction

Due to its overwhelming characteristics, the Internet of Things (IoT) has been used in almost every application domain, especially smart buildings and cities, healthcare, manufacturing, and agriculture. IoT consists of smart devices, i.e., things, with embedded sensing and communication modules to form a self-organized network and is very effective in automatically controlling various activities [1]. These devices are deployed in proximity to the underlined phenomenon and capture data after a defined time interval, which is shared with the nearest server via a secure wireless communication channel. As devices in IoT are densely deployed and communicate via wireless media, therefore, the authenticity of both parties, i.e., source and destination, is crucial and very challenging where both devices, that is, the server and member device, must ensure the authenticity of each other. Additionally, a message may be intercepted by an intruder device and, thus, every message should be encrypted with a secret key, i.e., λ, making it non-readable [2].

Device authentication is among the challenging issues with networks, especially IoTs, where devices, such as source and destination, transmit via wireless communication, which is highly susceptible to unauthorized access and interception of messages [3]. For verification of the intended device, secure handshake-enabled authentication mechanisms are adopted, where every device confirms the legitimacy of the intended device through a challenge that is encrypted using either a secret shared key, i.e., λ. In this mechanism, a series of encrypted messages, i.e., four, are transmitted from both parties, where every device tries to resolve the challenge of the intended device and responds with another message that contains the solution to the challenge along with its own, as shown in Fig. 1. In the literature, numerous mechanisms have been presented to resolve authentication issues, i.e., device-to-device and device-to-server, especially with IoT and other resource-constrained networking infrastructures. An efficient and effective authentication technique is presented by Hajny et al. [4] for the establishment of secure channels through an anonymous verification approach in IoT where the privacy of the intended device is preserved through proper utilization of anonymity.

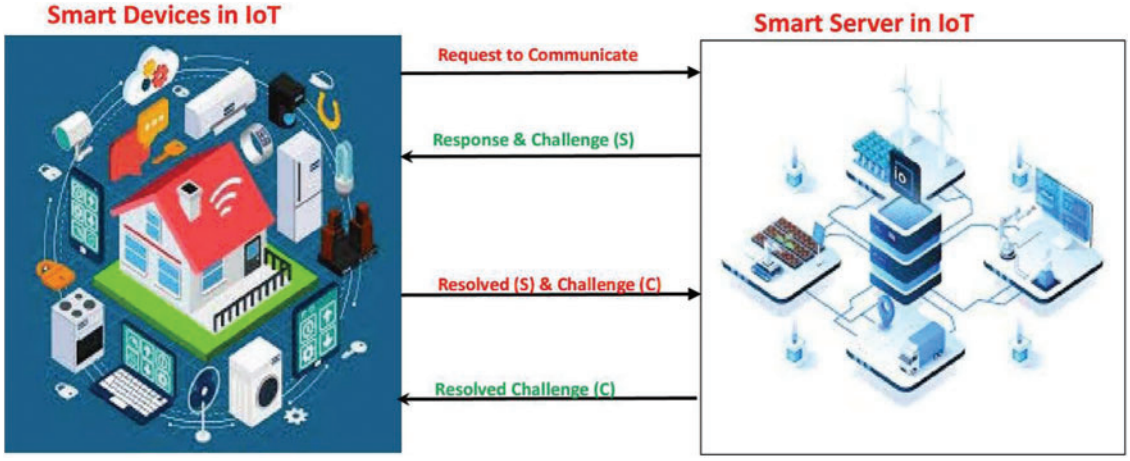

Figure 1: Generalized device-to-server authentication in the internet of things

Datagram Transport Layer Security (DTLS) has been used along with bit-wise exclusive OR (XOR) and hash functions to form a lightweight device-to-server authentication scheme where devices must ensure to carry out the intended process before the initiation of communication [5]. To ensure the integrity of the transmitted data in an open environment of wireless communication, a decentralized approach that is based on ledgers for the authentication of source and destination devices was

developed, where two different layers are utilized, i.e., (i) verification and ledger layers, respectively [6]. Anonymity and biometric approaches, which are very effective due to their strong security, were integrated to develop a trustworthy authentication system, preferably user-based, for the smart healthcare domain, such as the Internet of Medical Things. However, biometric-based approaches are feasible only where the intervention of human beings is required in every aspect of the system, which is not applicable in the majority of IoT application domains. Similarly, a privacy-preserving device-to-device authentication scheme, preferably lightweight, has been developed and implemented in the industrial Internet of Things (IIoT) environment. In this approach, devices and server modules are required to collect a secret ID and key, i.e., λ, from a trusted authority in the offline phase. These IDs and keys are then utilized in the authentication process to verify the legitimacy of the intended device in IIoT [7]. A hybrid and effective authentication scheme that has integrated MAC addresses and enhanced on-demand vector approaches makes sure that communication sessions are established only if both devices, i.e., member and server, are legitimate in the IoT [8]. Likewise, a three-factor oriented security scheme has been developed to ensure both anonymity and session key with available resources in wireless sensor networks and IoT [9]. Although these approaches have successfully resolved authenticity issues in particular infrastructures, none of them have considered location-based authentication schemes, which are common in IoT infrastructures. Another issue with existing approaches is device mobility, which is not supported in major schemes. For example, if a module could move from one region or location to another that does not fall within the coverage area of the respective server, then what would happen?

In this paper, the Asymmetric Encryption Scheme (AES) and media access control (MAC) are integrated to form a lightweight authentication approach that is specifically designed for IoT and other resource-limited networks. In this scheme, devices are required to be part of an activity, which is carried out in the offline phase, where registration and secret key-sharing processes are completed. Then, before the actual communication, every device C_i is required to be registered with the closest server module, especially the one deployed in that region. This registration process is subjected to verification of the device's ID and MaskID, which are provided by TA in the offline phase. Thus, a device C_i becomes a member of a server Sj only if its ID and MaskID are matched with already stored addresses, i.e., those shared by TA. Secondly, both devices, i.e., member and server, are required to verify the legitimacy of the other device through a competitive challenge that is generated and encrypted using its secret key λ. This challenge is resolvable only if the concerned device, i.e., C_i or S_j, has the respective secret shared key. The main contributions to this manuscript are given below:

- A lightweight device-to-server authentication approach, i.e., challenge-oriented, for the Internet of Things.
- A hybrid authentication approach is developed that is comprised of the 128-bit AES and MAC addresses of active devices in IoT and is suitable for other infrastructures as well.
- Authentication system with built-in support for mobility of devices, i.e., C_i, in IoT, where devices move from the coverage area of one server S_j to another S_{j+1}.

The remaining paper is organized as follows.

In the subsequent section, a comprehensive review of the most relevant literature, preferably those related to the authentication of devices and servers in the IoT, is presented. In Section 3, a detailed description of the proposed region-based authentication scheme is presented, whereas the system model of the proposed system is given in Section 4. In Section 5, a comprehensive analysis of the proposed region-based authentication scheme's performance in terms of various metrics is presented. Finally, concluding remarks are given.

2 Literature Review

Authenticity of devices, i.e., communication parties (member and server), is among the crucial aspects of the networking infrastructure in general and IoT in particular. The literature is quite bulky on addressing the device's authenticity issue, therefore, a comprehensive review of only those techniques, that are relevant to the proposed methodology, is presented. In 2016, Amin et al. [10] reported on the development of a two-tier authentication mechanism, which is based on three factors along with a bi-pairing technique, to safeguard an ongoing communication session, especially in the resource's constraint networking infrastructure. Although this approach was convincing, it was susceptible to offline password guesses and impersonation attacks. A secure certificate-enabled device-to-device authentication technique was developed to guarantee the authorization and integrity of devices & data, respectively. A security certificate and DTLS-enabled handshake approach, which is required to be completed before the actual communication, were introduced to safeguard the integrity & legitimacy of data and devices, respectively [11]. Similarly, an effective & secure scheme to update security keys, i.e., λ_i, has been designed and developed to resolve vulnerabilities, preferably related to various adversary attacks, in IoT networking infrastructure. This scheme was self-adoptive, a common property required in IoT, and other resources-constrained networks, to adjust itself accordingly, i.e., according to the security requirements of the domain [12]. A verification methodology, that is designed for wearable devices in the smart healthcare domain, has been reported, which is primarily based on a unique addressing methodology to differentiate legitimate devices from adversaries deployed in different parts. Furthermore, this scheme has ensured to preservation of anonymity of communicating parties, preferably through a secure session key, in the smart healthcare domain [13]. An epileptic curve and pre-shared key (PSK) enabled authentication technique has been developed and extensively tested in the resource-constrained environment of IoT networks. Additionally, this system has been designed such that it does not degrade the performance of the smart devices, especially in terms of services, i.e., data capturing and transmission [14]. An interesting lightweight methodology has been reported to preserve privacy and ensure the authenticity of communication parties in the smart healthcare domain. This methodology is based on lightweight functions, such as hash, to enable wearable devices to operate smoothly without compromising on security and privacy measures with the lowest possible communication and processing cost in smart healthcare infrastructures [15]. An updated version of the standard IPv6 security algorithm has been presented to enable its adoption on the Internet of Things where a unique 64-bit identity number is allocated to every member device along with a secure session key, i.e., to ensure privacy and security, especially in smart homes [16]. Likewise, a lightweight device-to-device authentication methodology, i.e., which is designed for the IoT, has been presented to secure ongoing communication sessions from fraudulent devices, i.e., adversaries, in the IoT networks. This scheme is secure against well-known intruder attacks, which are feasible in the open environment of IoT [17]. Apart from these, a COAP protocol-enabled authentication scheme has been developed to safeguard IoT networks from fraudulent users, i.e., adversaries or intruders, with minimum possible authenticity or legitimacy overheads [18]. To ensure the integrity of the transmitted data in the open environment of wireless communication, a decentralized approach, that is based on the ledger, for the authentication of source & destination devices was developed where two different layers are utilized, i.e., verification and ledger layers, respectively [6]. Likewise, an extensive analysis of the artificial intelligence-based secure and anonymous payment scheme has been carried out by Fragkos et al. [19] especially those schemes where the anonymity of source is preserved in E-cash. Anonymity and bio-metric approaches, which are very effective due to their strong security, were integrated to develop a trustworthy authentication system, preferably user-based, for the smart healthcare domain such as the Internet of Medical Things. However, bio-metric-based approaches are feasible only where the

intervention of human beings is required in every aspect of the system which is not applicable in the majority of IoT application domains. Similarly, a privacy-preserving device-to-device authentication scheme, preferably lightweight, has been developed and implemented in the industrial Internet of Things environment (IIoTs). In this approach, devices and server modules are required to collect secret IDs and keys, i.e., λ, from a trusted authority in the offline phase. These IDs and keys are then utilized in the authentication process to verify the legitimacy of the intended device in IIoT [20]. A hybrid and effective authentication scheme, which has integrated MAC address & enhanced on-demand vector approaches, to make sure that communication sessions are established only if both devices, i.e., member and server, are legitimate in IoT [8]. Likewise, a three-factor oriented security scheme has been developed to ensure both anonymity and session key with available resources in the wireless sensor networks and IoT [9].

Likewise, a three-phase-enabled authentication scheme has been introduced by Aziz et al. [21], where devices are bound to be registered with the authentication server and vice versa. In the authentication phase, a secret key along with other relevant information is exchanged to ensure the authenticity of both the source and destination module in the IoT network. Even though these schemes are effective in ensuring a secure communication infrastructure, each of these schemes is susceptible to adversary attacks such as denial of service, Replay Attacks, and Edge or Server Impersonation attacks. Moreover, most of these approaches are designed for specific environments or overlay complexes, so their realization is very hard. Thirdly, existing approaches have not considered an important security aspect which is the location of the respective device or server module. Finally, existing approaches do not support the mobility of either member devices or server modules, which is desperately required in different application domains such as smart hospitals where a patient is moved from one ward to another. Therefore, a lightweight, that is suitable for any resource constraint devices, and a secure authentication approach is required to be developed which not is prunes against well-known security or authenticity breaches but is equally effective in terms of minimum processing and communication cost.

3 Proposed Region-Based Authentication Scheme for the Internet of Things

The authentication scheme bounds server devices S_j to maintain secret keys, such as λ_i, information, and IDs of all client devices C_i, which is feasible for limited or small IoT infrastructures. To resolve this issue, a region-based authentication scheme is presented in this section. This scheme bounds server devices to keep a record of every client device ID, but secret keys λ_i are maintained region-wise that is each region has a specific secret key λ_i. A client device interested in communication initiates a request message that contains the device-ID and region-ID, i.e., reference point, and encrypts this message with a shared secret key λ_i. The concerned server device S_j decrypts this message with a shared secret key λ_i, which is performed in the offline phase and responds with a server challenge as described in the Eq. (1) with an appended client device ID.

$$\gamma_{server-payload} = AES\left(\lambda_i, \psi_{resultant} | \eta_{server} | C_{ID}\right) \tag{1}$$

A client device C_i with a matching ID generates a client challenge for the concerned server device S_j using an Eq. (1), while other devices ignore this message. If a server challenge is collected by an intruder device D_i, it needs 2^{128} iterations to decrypt this message, as λ_i is known only to both server device S_j and client devices C_i which reside in that region. The server device S_j decrypts this message, which is encrypted using session key μ_i, and generates an authentication payload that is encrypted with shared secret key λ_i and broadcasts it. The concerned client device decrypts this message and confirms the server's authenticity by matching its η_{client} with that embedded in the message. This approach not

only preserves the authenticity of both server and client devices with a minimum possible set of secret keys λ_i but saves considerable resources, particularly processing and response time.

Theorem 1: A session initiation process is generated by a device C_i **iff** $C_i \in$ Mem (S_j).

Proof: To create a session initiation request, a secret key λ_i is needed, which is known only to the member client devices such as C_1, C_2, and C_3, $C_n \in$ Mem (S_j). If an intruder device D_i, somehow, replicates the ID of a $C_i \in$ Mem (S_j), that is either generated or accessed, but still, requires the secret key λ_i of that region, which needs sophisticated and complex hacking techniques to get it, which is not possible with a resource-limited D_i.

Conversely, if a member device C_i initiates a session establishment request $Msg_{request}$ and encrypts it with its secret key λ_i, then server S_j of that region can decrypt this $Msg_{request}$ and generate a response message as server challenge $\Upsilon_{server\text{-}challenge}$. Additionally, if client device C_{i+1}, that is $C_i \notin$ Mem (S_j), initiates a session establishment request $Msg_{request}$ and decrypts it with its λ_i, then it is ignored by the server S_j as $C_i \notin$ Mem (S_{j}). Hence, a session request is initiated by a client device such that $C_i \in$ Mem of that S_j, where Mem = $\{C_1, C_2, C_3, \cdots, C_n\}$.

4 System Model of the Proposed Region-Based Scheme

The authentication process of both devices, i.e., ordinary (sender or source) & server (destination or receiver) modules, should be completed, preferably through encryption & description within a defined time frame, before the initiation of actual communication in IoT. The notations used in the proposed setup are given below in the Table 1.

Table 1: The notations used in the proposed setup

Acronym	Description
C_i	Authentic member device
S_j	Authentic server device
PT	Plain text
CT	Cipher text
λ_i	Secret shared key
$\Upsilon_{device\text{-}challenge}$	Challenge generated by member device to ensure the authenticity of the respective server module
$\Upsilon_{server\text{-}challenge}$	Challenge generated by the server to ensure the authenticity of the respective member device
Mem ()	Class of member devices shared in the offline phase
$Msg_{request}$	Message generated by requesting device
η	Random number used in the authentication process
μ_i	Session key of member device
μ_j	Session key of the server module
MaskID	Use to hide the identity of the requesting device
TA	Trusted authority
MAC	Media access control
||	Appending information
ΔT	Time delimiter

(Continued)

Table 1 (continued)

Acronym	Description
$\psi\ (C_i)$	Random number generated by a member device
T_{RX}	Receiving time
T_{TX}	Transmission time
$\oplus$	Exclusive OR operation

In the proposed setup, which is specifically designed for IoT, devices are broadly divided into three groups as given below:

1. Ordinary device, which is represented by C_i, and deployed in the vicinity of the respective phenomenon. Every device C_i has a unique ID, which is the MAC address in this case.
2. Server module, represented by S_j, and responsible for receiving captured data values from the authentic devices such that informed decisions are made.
3. Trusted devices (TA), represented as TA_i, can allocate mask IDs with the intended server modules and devices.

It is important to note that TA plays a vital role in the proposed authentication setup for IoT, where every device, i.e., ordinary C_i or server S_j, is required to be registered with TA preferably in an offline phase. To ensure this, every device, i.e., ordinary C_i or server S_j, is required to be registered through the defined procedure by the respective TA where the device's IDs, i.e., MAC address in this case, are shared, preferably in encrypted form. Secondly, mask IDs of these devices, i.e., ordinary C_i or server S_j, are computed using the given Eqs. (2) and (3), respectively.

$$Mask_{ID}(C_i) = hash(MAC_i||AES_i) \tag{2}$$

where C_i represents the complete set of legitimate active devices in IoT, which are divided into different regions.

$$Mask_{ID}(S_j) = hash(MAC_j||AES_j) \tag{3}$$

where i and j are used to represent the total set of active devices, i.e., C_i, and servers, i.e., S_j, in IoT. These mask IDs are generated for every individual device, i.e., ordinary C_i or server S_j, by TA in the offline phase and shared with the respective devices in IoT. Apart from it, every device C_i & server module S_j has a 128-bit unique key, i.e., λ_i, that is shared by TA through the encrypted message in the offline phase where the probability of intruder's entry is almost negligible as IoT has not operational yet. Moreover, every device, i.e., ordinary C_i or server S_j, is assumed to be a potential candidate for legitimate devices in the offline phase. These processes are described in detail one by one below.

Additionally, the proposed authentication model could be implementable in scenarios where multiple servers, i.e., preferably those located in different infrastructures, are deployed to provide a better communication environment, i.e., with minimum information or packet loss ratio, along with built-in support for mobile devices in the IoT. Secondly, a device Ci should be connected, i.e., will be able to communicate and share captured data values, to a single server, i.e., S_j, at a particular time interval in the IoT. However, if somehow its position is changed, i.e., moved to another location, i.e., in a smart healthcare environment where a patient is moved from one ward to another, then it requires repeating the authentication process with the nearest server S_{j+1}. Additionally, before its

movement, this device must inform the respective server S_j through a short message that it is moving from the coverage area. The respective server module must remove this device from the authentic devices list and, thus, it becomes just a member device. In the future, if this device is interested in communicating with the intended server module, then it must authenticate itself through the described process. Moreover, location information is very critical in the proposed infrastructure as it is used as one of the security measures. As every server module has location information, i.e., reference point, about its member devices, which is shared in the off-line phase, if somehow an authentic device is compromised, then the server module would be able to differentiate the intruder device, which pretends to be an authentic device, by utilizing location information that is the coordinates of the devices. Thus, the proposed approach is compromised only if the intruder device not only knows the MAC address, MaskID, and secret shared key along with the location information of the compromised device, then it will be able to start a proper communication session with the respective server module. However, attaining all these security metrics is far beyond the operational capability of the resource-limited device. However, if we assume that the adversary has a highly sophisticated system with exceptional processing facilities, then it is still very difficult to capture and replicate that information, especially location coordinates, that is reference point, along with ΔT.

The proposed model could easily be extensible and implementable in almost every domain of IoT with minor preferably negligible modifications and, thus, it does not suffer from the scalability issue that is related to most of the existing state-of-the-art approaches. As it is a multiple server-oriented model, therefore, could be extended according to the application requirements, especially in the IoT domain. Secondly, implementation of the proposed model is very easy as it is based on a hierarchical approach, a common approach in IoT infrastructure where devices are bound to share information with the nearest server module through direct communication. Finally, the proposed approach is an ideal solution for the future IoT infrastructure where it is highly likely that devices or server modules could be mobile devices and this service is already available in the proposed model.

4.1 Registration Phase: Devices and Servers with TA

In this phase, every device, i.e., ordinary C_i or server S_j, shares its MAC (media access control) address with the respective trusted authority (TA) with an embedded secret key, i.e., μ_i, used to encrypt messages in the offline phase. TA is responsible for generating mask IDs, i.e., $MaskID_i$ & $MaskID_j$, for every device, i.e., ordinary C_i or server S_j, using Eqs. (2) and (3) respectively and shared with these devices through encrypted messages in IoT. Additionally, TA shares a unique secret key, i.e., λ_i & λ_j, with every device and server module, respectively. Initially, every device, i.e., ordinary C_i or server S_j, sends its MAC address, preferably in encrypted form, to the respective TA. The concerned TA deciphers this message with his unique secret key λ and generates Mask ID using either Eqs. (2) or (3). These mask IDs are shared with the concerned device, i.e., ordinary C_i or server S_j, using an encrypted message, which is carried out through 128-bit AES-enabled secret key λ. As soon as the registration of all devices, i.e., ordinary C_i or server S_j, is completed, then TA shares a complete list of authentic devices, $C_{1\ldots i}$, with server S_j and vice versa. Thus, every device, i.e., ordinary C_i or server S_j, has a list of legitimate devices, i.e., ordinary C_i or server S_j, in IoT.

In the next phase, every device, C_i, needs to be a registered member of the nearest server module, S_j. For this purpose, a device, C_i, which is interested in communicating with a respective server module S_j, generates a random nonce ψ_i using Eq. (4) and encrypts it with a secret key, λ_i using a 128-bit AES encryption scheme.

$$\psi(C_i) = rand(num) \tag{4}$$

In addition to it, the device, C_i, computes mask ID, which is a hybrid of random nonce η and mask ID assigned to it in the offline phase. Then, device C_i generates a cipher text message where the payload consists of MAC, mask ID, and random nonce ψ and sends it to the respective server module S_j, which is deployed in the region. As this message is sent in encrypted form, therefore, adversary A_k, if located in the vicinity, will not be able to view its contents.

At server module S_j, the encrypted message is converted into plain text using a secret key, λ_j that is shared by TA in the offline phase. MAC address and mask ID of requesting device C_i are verified by checking their entry in the authentic devices list, which is shared by TA in the offline phase. If the required attributes, i.e., MAC address & mask ID, are available in the stored list, then requesting device C_i is added to the member devices class of the respective server S_j. However, if either one of these attributes or both are missing, then the server assumes that the requesting device is not trustworthy and may be a potential intruder device. Thus, blacklist it and share this information with the nearest server module, S_{j+1} as well.

Theorem 2. A legitimate or registered device C_i, not an intruder A_k, can become a member device with the nearest server S_j.

Proof. Every device C_i gets MaskID and 128-bit AES-based secret key, λ_i, from TA in the offline phase and the same, i.e., maskID & MAC address of authentic devices, is shared by TA with every server module S_j in the offline phase where the probability of possible entry of potential adversaries or intruder devices A_k is negligible. However, if we assume that an adversary A_k has generated a membership request message, somehow with the same attributes, using its secret key λ_k, and sent it to the nearest server S_j. It will be rejected as the concerned secret key, λ_k, is not known to the server module and, thus, will not be able to convert this message into plain text which is a clear indication that the request has come from the adversary. However, if we further assume that adversary A_k has intercepted this message and forwarded an updated version of the original message to the respective server S_j. The server module proceeds to verify the authenticity of the requesting device A_k by (i) searching both attributes, i.e., MAC address and MaskID, in the stored database & expected arrival time ΔT, which is described in Eq. (5).

$$\triangle T = T_{RX} - T_{TX} \tag{5}$$

where T_{TX} and T_{RX} represent the transmission and reception time of the concerned message, it is important to note that intruder Ak cannot convert the intercepted message into plain text, modify it, and then transmit it to the respective server S_j. Moreover, as this device A_k did not participate in the offline phase, therefore, its MAC address and MaskID will not be available where information about authentic devices is stored, i.e., $\text{MaskID}(A_k) \notin \text{MaskID}_{1\ldots\ldots n}$ and $\lambda_k \notin \lambda_{1,2,3\ldots n}$. Hence, it will be identified as an adversary.

Conversely, if this request message is sent by a legitimate device C_i, then as this C_i was part of the offline phase, thus, its information, i.e., MAC address & MaskID, will be stored with the respective server S_j. Therefore, a match will be found, i.e., $\$\text{MaskID}\ (C_i) \in \text{MaskID}_{1,2,3\ldots n}$ and $\lambda_i \in \lambda_{1,2,3\ldots n}$, as soon as this information is searched in the database of any server S_j in IoT. Thus, requesting device C_i is identified as an authentic device and added to the member devices class of the respective server module. Thus, a legitimate or registered device C_i, not an intruder A_k, can become a member device with the nearest server S_j.

Furthermore, secret keys $\lambda_{1,2,3\ldots n}$ which are assigned to potential authentic devices $C_{1,2,3\ldots n}$, are formed through equality principles which are given below:

1. λ_i of a device $C_i = \text{PT} \oplus \text{Round}_{0\text{–}9} \oplus \text{Add}_{\text{rkey}} \oplus \text{CT}$.

2. λ‘$_i$ of a device $C_i = PT \oplus Round‘_{0\text{–}9} \oplus Add_{rkey} \oplus CT$

where PT & CT represent plain and cipher text, respectively. Similarly, rkey and round are utilized to represent generated secret keys in round a process to continuously compute unique keys, respectively. In this way, every device C_i is registered with server S_j, which is deployed in the respective location, especially with maximum receive signal strength indicators. Additionally, every server S_j has a list of both member devices C_i, i.e., those sent a request to be a registered member such that $\forall_{i=1}^{n} C_i \in$ (*Authentic* & ***Memberclass***) and non-member devices C_l such that $\forall_{l=1}^{m} C_l \in$ *Authentic*.

Apart from that, the proposed authentication model has built-in support for the mobility of authentic devices from the coverage area of one server S_j to another S_{j+1} without changing the technological infrastructure and additional resources of IoT. If a device C_i migrates from the coverage area of an S_j to another S_{j+1}, then it just needs to repeat the procedure for registration with another server module.

4.2 Authentication Phase: Ordinary Devices & Server

This phase is dedicated to the most crucial aspect of the proposed authentication scheme, i.e., authenticity verification of both parties that is both requesting device C_i & respective server S_j, preferably in the presence of intruder devices A_k in IoT. Although both devices, i.e., C_i & S_j, have a stored list, which is shared by TA in the offline phase as described above, of legitimate devices in IoT, an intruder device Ak may likely pretend itself as a legitimate device. Thus, to ensure that only legitimate devices, i.e., C_i & S_j, are permitted to communicate, the authentication process is mandatory to be carried out before the sharing of information or other resources. In the proposed setup, the authentication process of both parties is broadly divided into four (04) subcategories, i.e., (i) Challenge of the Requesting Device C_i (ii) Challenge of the Respective Server S_j (iii) Requesting Device's C_i Authenticity, and (iv) Respective Server's S_j Authenticity.

4.2.1 Challenge of the Requesting Device C_i for Respective Server Module

In the proposed setup, if a device C_i has data to be shared with the respective server, however, to preserve the integrity of captured data, the authenticity of the respective server S_j must be confirmed & verified before the information sharing process in IoT. For this purpose, the requesting device generates a cipher text message using 128-bit AES-based encryption, i.e., a challenge for the respective server S_j and easily resolvable if secret key λ_j is available, consisting of its ID (MAC address, $\psi_{Ci\,\&Sj}$, and time), $Chall_{server}$ (Exclusive OR of random nonce & ID), and $Session_i$, i.e., $\psi_{session}$. Random nonce η_{device} is formed through a specifically designed random number function as depicted in Eq. (6).

$$\eta_i(C_i = Pseudo - Random()) \tag{6}$$

Secondly, as both devices, i.e., C_i & S_j, are likely to communicate for an entire session, therefore, a one-time usable number, i.e., $\psi_{Ci\,\&Sj}$, is generated using Eq. (7). For this purpose, an exclusive OR operation is applied on both the η_i and MAC address of S_j, which is already shared by TA in the offline phase.

$$\psi_{C_i \& S_j} = \eta_{C_i} \oplus \text{MAC}(S_j) \tag{7}$$

Finally, the challenge for the respective server, i.e., S_j, is formed, which consists of ID, $\psi_{Ci\,\&\,Sj}$, MaskID, and T_1 (Transmission time), and converted into cipher text through 128-bit AES-Enabled

encryption as depicted in Eq. (8).

$$\gamma_{sc} = AES(\psi_{C_i \& S_j} \oplus MAC_{C_i} || T_1) \tag{8}$$

where variable sc has been utilized for the effective representation of the source device's payload and T_1 is the actual time of message transmission. This message is very challenging and informative as it contains valuable information required for the verification of both devices in IoT.

4.2.2 Challenge of the Server S_j for the Requesting Device

As soon as the cipher text message transmitted by device C_i is received by the respective server S_j, then it converts it into plain text using a secret key, i.e., λ_j, which is shared by TA in the offline phase and extracts information stored in the message payload. Initially, server S_j confirms the authenticity of device C_i through a comprehensive methodology where the MAC address of this device is searched in the stored address, which is verified through Eq. (9). This device, i.e., C_i, is considered as authentic if and only if the MAC(C_i) ∈ stored (MAC_{reg}).

$$Authentic_{C_i} = \exists_i^n MAC_i \in Class(MAC_j) \tag{9}$$

Secondly, the authenticity of this device C_i is further confirmed through another metric, i.e., message delivery time, which is computed using Eq. (10) as given below:

$$\Delta T = T_2 - T_1 \tag{10}$$

where T_1 & T_2 represent transmission and delivery time respectively. Secondly, the CID_i of the respective source device is matched against the legitimate devices using the Eq. (9).

Alternatively, if the MAC address of device C_i has not been found in the stored database, then the respective server S_j assumes that requesting device may be a potential intruder device and, thus, add its MAC address to the blacklisted class. Similarly, if the expected packet delivery time does not fall within the permitted bounds, then this scenario is also considered a potential security breach, and the device's MAC address is added to the blacklisted class. If device C_i is verified as authentic, then server S_j generates a challenge for the respective device to further confirm its legitimacy. To ensure this, random nonce extracted from the cipher message is used, which is an indicator for the respective device C_i that S_j is an authentic server and can be trusted for sharing information.

This number, i.e., $\eta_i(C_i)$, along with its MAC address is passed through exclusive OR operation to generate 128 bits $\psi_{Ci\,\&Sj}$ as given below in Eq. (11).

$$\psi_{C_i \& S_j} = MAC_j \oplus \eta_i(C_i) \tag{11}$$

The idea behind using the same $\eta_i(C_i)$, i.e., in the server challenge, is to ensure the requesting device C_i that message has been successfully converted into plain text by a trusted device, which is possible only if receive the module, i.e., S_j in this case, has the secret shared key, λ_i, which is shared by TA during the offline phase. Sever S_j then generates its challenge, i.e., γ_{sc} for the respective device C_i by passing the result of Eq. (11) and its MAC address of C_i through exclusive OR operation as given below in Eq. (12).

$$\gamma_{sc} = AES(\psi_{C_i \& S_j} \oplus MAC_{C_i} || T_3) \tag{12}$$

where sc & T_3 represent the message payload and transmission time of the server modules, respectively.

Theorem 2. Cipher text message, i.e., γ_{sc}, is convertible into the respective plain text **iff** C_i OR A_k has one of the unique secret keys, i.e., $\lambda_{1\ldots i}$.

Proof. If we assumed that this cipher message, i.e., server's challenge γ_{sc}, is somehow intercepted by a non-member device, i.e., potential intruder A_k, that is placed in closed proximity of server S_j, then it will try to read & alter its contents, but to do so, A_k requires to convert this message into plain text, which is possible only, if it has one of the secret key, i.e., $\lambda_{1\ldots i}$. As it is an intruder, therefore, it will utilize various keys, i.e., $\lambda_{1\ldots k}$, through a head & trial procedure that is function f(k), where success probability is directly proportional to the set of secret keys, i.e., $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \ldots \lambda_n$, such as given in Eq. (13).

$$f(y) = \exists_{i=1}^{k}[\because \lambda_k \in \lambda_1, \lambda_2, \lambda_3, \ldots \lambda_n \,\&\, MAC(A_k) \epsilon StoredMAC] \tag{13}$$

For this purpose, intruder device $_{Ak}$ is required to generate a random key, i.e., λ_k, which must be identical to one of the legitimate secret keys, i.e., $\lambda_1, \lambda_2, \lambda_3, \ldots \lambda_n$, through a mathematical Eq. (14) as given below:

$$\lambda_k = Head \,\&\, Trials\left[statistics\left(\lambda_i \oplus \lambda_j\right)\right] OR\, Random_{key}(AES - 128) \tag{14}$$

First, it is hard for the intruder device $_{Ak}$ to convert this message into plain text form using the head & trial technique as described in Eq. (14), but it is a time-consuming and lengthy process. However, if we further assume that somehow, it did, then there is another parameter (timestamp), i.e., ΔT, that is used to verify the integrity of the intercepted message. Converting a 128-bit AES-based cipher text message into plain is a lengthy process and, thus, it is very hard for intruder A_k to deceive the respective server S_j with its interpreted & updated message. As the updated message will have a different timestamp than what is expected from the legitimate device, therefore, message will be discarded by server S_j.

Conversely, if the cipher message is intercepted by an authentic member device C_i, then it will convert this message, i.e., γ_{sc}, into proper plain text using its secret key λ_i, which is shared by TA in the offline phase. Additionally, an authentic device can generate a response message within the stipulated time interval, i.e., ΔT. Thus, it proves that a 128-bit AES-based cipher text message, i.e., $\$\gamma_{sp}$, is convertible into plain text form only, if the device C_i has a legitimate secret key, i.e., λ_i.

4.2.3 Device Authenticity through Resolving Server Challenge

The authentication process of the requesting device is subjected to registration of **(i)** MAC address **(ii)** MaskID, **(iii)** ΔT should be as expected, **(iv)** successful conversion of server challenge, i.e., γ_{S_j}, and **(v)** generate a challenge, i.e., γ_{C_i}, for the respective server within the stipulated time interval. Thus, server S_j verifies the authenticity of device C_i using Eq. (15).

$$Authentic_{C_i} = MAC_i \in RegisteredMAC(S_j) \;\; \& \;\; MaskID \in Registered\, MaskID(S_j) \;\; \& \;\; \psi_{S_j} \in \gamma_{C_i} \;\; \& \;\; \Delta T \;\; is \;\; as \;\; expected \tag{15}$$

In response to server challenge, i.e., γ_{S_j}, the concerned device C_i generates a cipher text message, i.e., with its 128-bit secret key, which contains an exact copy of the random number, i.e., ψ_{S_j}, is server challenge, i.e., γ_{S_j}. Additionally, this message is ciphered through one of the legitimate secret keys, i.e., $\lambda_i \in \lambda_1, \lambda_2, \lambda_3, \ldots \lambda_n$, which are already registered with server S_j.

Secondly, if server S_j received a challenge from the respective device C_i, which contains a random number, i.e., ψ_{S_j}, n it is an indication that device C_i requesting authentication is a legitimate device with its secret key, i.e., λ_i. Thirdly, if server S_j converts & resolves the challenge of the requesting device C_i,

then it points out the legitimacy of C_i, as server S_j can convert only those messages to plain text form, which are ciphered through one of the legitimate secret keys, i.e., $\lambda_i \in \lambda_1, \lambda_2, \lambda_3, \ldots \lambda_n$.

Theorem 3. The challenge of server S_j, i.e., γ_{S_j}, is resolved if and only if, device C_i is legitimate with authentic key, i.e., λ_i.

Proof. Let us consider that, the challenge of server S_j, i.e., γ_{S_j}, is intercepted by an adversary A_k, deployed somewhere in the middle of S_j & C_i and begins to convert this challenge, i.e., in cipher text form using its secret key, i.e., λ_k, which is generated using Eq. (16).

$$\lambda_k = Head \;\; \& \;\; Trials\left[Probabilistic\left(\lambda_{1,2,3,\ldots n} \oplus \lambda_{1,2,3,\ldots m}\right)\right] OR\, Random(AES-128) \tag{16}$$

In addition to the Eq. (16), adversary A_k may utilize a probabilistic function, i.e., G (x), where knowledge about the nature of the legitimate devices, C_i, is used to find an actual correlation between MAC address and secret keys (expected not actual) and it is based on head & trial method, i.e., λ_i, through Eq. (17) as given below:

$$G(x) = PROB\left(MAC(C_1), (\lambda_{1,2,3,\ldots k}), MAC(C_2), (\lambda_{1,2,3,\ldots k}), \ldots MAC(C_n), (\lambda_{1,2,3,\ldots k})\right) \tag{17}$$

To carry out these tasks, adversary A_k should closely monitor every ongoing communication(s) with the respective server S_j and will try to find certain patterns. However, to convert a 128-bit AES-based encrypted message into plain text form, a secret shared key, i.e., λ_i, is required, which is possible to break only, if A_k applies 2^{128} keys one after another. Keeping in mind the complexity of the 128-bit AES-based secret key, i.e., λ_i, it is not possible for adversary A_k, particularly by considering the limited processing power of these devices, to resolve the server's S_j challenge within the specified time frame.

Conversely, if a challenge, i.e., γ_{S_j}, generated by authentic device S_j is intercepted by the intended device C_i\$, then C_i uses its key λ_i to convert it into proper plain text format, extract information from the payload and will send a reply message, η_{sc}, within the expected time frame as key, i.e., $\lambda_i \in \lambda_1, \lambda_2, \lambda_3, \ldots \lambda_n$, which are shared during the offline phase. Thus, the challenge of server S_j is resolved if and only if, device C_i is legitimate with an authentic key, i.e., λ_i.

4.2.4 Server's Authenticity through Resolving Device's Challenge

In the previous phase, server S_j has verified the legitimacy of the requesting device S_j through server challenge, i.e., γ_{S_j}, which is resolved by the concerned, however, the authenticity of S_j is still questionable and requires to be confirmed by the respective device C_i. For this purpose, device C_i uses (i) MAC address and (ii) MaskID of the respective server by reading the content of the message's payload, i.e., γ_{S_j}. However, to further verify it, device C_i creates a challenge, i.e., γ_{C_i} in response to the server S_j challenge as depicted in Eq. (18) and send it in the cipher text form. Additionally, device C_i embeds a random number, i.e., η_{C_i}, in the message's payload and expects to receive a reply within the expected time frame.

$$\gamma_{sc} = AES[MAC\left(S_j\right) \oplus \psi_{C_i \& S_j} || T_3] \tag{18}$$

where $\psi_{C_i \& S_j}$ is generated using Eq. (19).

$$\psi_{C_i \& S_j} = MAC_j \oplus \eta_i(C_i) \tag{19}$$

This message is transmitted in cipher text form, i.e., encrypted with 128-bit AES-based key λ_i, and, thus, covert-able to plain text form only if intercepting device, i.e., legitimate S_j or intruder A_k, has the required secret key, i.e., λ_j. Now, this message may be intercepted by an intruder device A_k or a legitimate server S_j, both will try to extract information contained in the payload section of the

message, i.e., γ_{C_i}. If an adversary intercepts this message, i.e., γ_{C_i}, and tries to extract information contained in the payload section. However, γ_{C_i} is required to be converted to plain text form, which is applicable only if the secret key, i.e., λ_i, is available. As adversary A_k does not have the secret shared key, i.e., λ_i, therefore, it may try to convert this message into plain text using the head & trail procedure, which is time-consuming and most probably less successful as far as security of 1128-bit AES-based encryption is concerned. Through the successful exchange of these messages, i.e., i.e., γ_{C_i} & i.e., γ_{S_j}, authenticity of both modules is verified.

Finally, the proposed authentication model ensures that a device can move from the coverage area of the respective server to another. However, before its movement, the device must inform the concerned server about its planned movement, preferably through a short message that is encrypted using its secret key. As soon as the server module receives this message, it will change the status of this device from authentic to member device only. Secondly, when this device enters the coverage area of another server module, then it will initiate the respective authentication process as described above. A possible case study is a patient admitted in the respective ward-A of a smart IoT-enabled hospital where patients are attached to wearable devices to monitor their status. These wearable devices are directly attached to the respective server module deployed in the concerned ward or building and could allow these wearable devices after a rigorous authentication process. Now, if a patient needs a CT scan OR other tests that are not possible in the same building and, thus, patients could be taken to the respective block. In this case, wearable devices attached to the patient's body leave a message to the concerned server and when these devices enter the coverage area of another server, the authentication process is triggered again to become authentic members of the new server module.

However, the proposed model is well equipped against well-known intruder attacks such as denial of services, man in the middle, reply attacks, device and server impersonation attacks, and masquerading attacks. However, the security and applicability of the proposed authentication model are solely based on the offline line phase, which is assumed to be non-accessible for the intruder module. However, if an adversary attends the offline phase, somehow, then it will be assumed as a legitimate device as it will take part in all activities that are carried out. Moreover, it is important to note that TA does not confirm the legitimacy of the request, especially in the offline phase as it has assumed that this phase is secure from adversary attacks.

5 Simulation and Analytical Results

To verify various claims of the proposed authentication scheme, it is implemented in NS-2, an opensource simulation software, where every device, i.e., C_i, is assumed to communicate with the nearest server module, especially with minimum possible distance or maximum received signal strength indicator (RSSI) in IoT. The proposed lightweight authentication along with existing state-of-the-art approaches are developed and comparison is carried out in terms of approximate processing and communication time intervals. Apart from that, these schemes were thoroughly checked in terms of bandwidth utilization and additional overheads. A detailed discussion of these parameters is provided in the following subsections.

5.1 Processing Time of the Proposed and Existing Techniques in the Internet of Things

Processing time is defined as the time taken by a particular methodology or approach to verify the authenticity of the communication party, i.e., the source that is device C_i and destination (server S_j) in this case, in IoT infrastructure. Therefore, a newly developed authentication approach is assumed to be acceptable if and only if it has achieved the minimum possible processing time than existing

approaches preferably under similar environmental, i.e., devices, and infrastructure of IoT. To verify this assumption, a comparative analysis of both proposed and existing authentication approaches is reported in Table 2 where exclusive R (XOR) and hash functions are used as the evaluating metrics. From Table 2, we have observed that the proposed lightweight approach verifies the legitimacy of both parties, i.e., source C_i & destination S_j, with minimum possible processing overheads. Additionally, the proposed authentication has built-in support for mobile devices, which is very common in different application areas such as hospitals, where device C_i must be authentic and has taken part in the offline phase whereas the majority of the existing approaches do not support mobility of either device or server in the active IoT.

Table 2: Comparison of the computational cost overhead

Schemes	User/Client	Device C_i	Server S_j	Total cost
Proposed location-aware	–	$2T_h + 2T_{XOR}$	$2T_h + 3T_{XOR}$	$4T_h + 5T_{XOR}$
Liu et al. [22]	$3T_h + 3T_{XOR}$	–	$4T_h + 12T_{XOR}$	$7T_h + 19T_{XOR}$
Gope et al. [23]	$3T_h + T_{XOR}$	–	$9T_h + 4T_{XOR}$	$12T_h + 5T_{XOR}$
Abdelshafy et al. [24]	$5T_h + 5T_{XOR}$	$2T_h + 1T_{XOR}$	$2T_h + 6T_{XOR}$	$9T_h + 12T_{XOR}$
Gupta et al. [25]	$7T_h + 4T_{XOR}$	$4T_h + 4T_{XOR}$	$5T_h + 3T_{XOR}$	$16T_h + 11T_{XOR}$
Makhalouf et al. [26]	–	$2T_h + 6T_{XOR}$	$7T_h + 7T_{XOR}$	$9T_h + 13T_{XOR}$
Hasan et al. [27]	$2T_h + 6T_{XOR}$	$2T_h + 5T_{XOR}$	$7T_h + 7T_{XOR}$	$11T_h + 18T_{XOR}$

5.2 Proposed and Existing Scheme's Communication Cost Overhead

Apart from the processing cost, these schemes, i.e., proposed AES-enabled lightweight and existing approaches, are thoroughly evaluated in terms of bandwidth requirements, which are assumed to be among the top priority resources in every networking infrastructure, to ensure timely transmission of the respective information in IoT. From this perspective, an authentication approach, i.e., existing or proposed, is assumed to be the best choice if its bandwidth requirements are at the minimum possible level, however, it should not compromise on overall speed and performance of the IoT. A comparison of these approaches in terms of communication or transmission overhead is depicted in Table 3 where the total number of bits transmitted by technique are shown especially those required to complete the authentication process between device and server. From the results, we conclude that the proposed lightweight authentication is a more suitable candidate than existing approaches for IoT environments where communication parties are required to be verified first. Secondly, the proposed scheme supports both static and mobile devices, which is very common in the IoT.

Table 3: Comparison of the communication cost overhead

Schemes	No. of messages	Bits
Proposed location-aware	04	512
Khan et al. [20]	06	1,536
Liu et al. [22]	06	30,620

(Continued)

Table 3 (continued)

Schemes	No. of messages	Bits
Gope et al. [23]	04	31,184
Abdelshafy et al. [24]	05	24,546
Gupta et al. [25]	05	3,038
Makhalouf et al. [26]	05	6,144
Hasan et al. [27]	06	32,000

5.3 Memory or Storage Overhead Metric in IoT

Memory or storage is an equally important evaluation metric especially when the resource-limited nature of member devices in IoT is concerned as it is mandatory for the smooth operation of the underlined networks. In Table 4, a detailed comparative analysis of the proposed lightweight and existing state-of-the-art methodologies is presented which shows that the proposed scheme has outperformed. In the proposed scheme, a device C_i is required to store its MastkID & secret shared key, i.e., λ, which are received from the TA in the offline phase. Additionally, minor space is required to store a list of authentic servers, i.e., S_j, which is required in scenarios where devices are mobile.

Table 4: Comparison of the memory or storage cost overhead

Schemes	Devices	Servers/Gateway
Proposed location-aware	Mask$ID_j + \lambda_j$	Mask$ID_i + \lambda_i$
Khan et al. [20]	$ID_j + \lambda_j + MID_j$	$ID_i + \lambda_i + MID_i$
Lie et al. [28]	$ID_i + \lambda_i + ID_G + \lambda_G$	$n\{ID_i + \lambda_i\} + m\{ID_G + \lambda_G\}$
Gope et al. [23]	–	$x_{id} + TS_{ug} + \omega + K_{ug} + Ts_{ugnew} + Sn_{id_i}^{new} + K_{gs_i}^{new}$
Gupta et al. [25]	$e_i + f_j + x_i + MI_u + MGID_i$	$Z_j + x_i + MGID_j$
Makhalouf et al. [26]	–	$PI_{RSU} + PK_{TA} + V_i + K_{vi} + ID_{vi}$

5.4 Security Analysis in Terms of Numerous Possible Attacks in IoT

Finally, the proposed lightweight authentication approach is thoroughly checked against well-known adversary attacks especially those that are linked to the IoT infrastructure, and the resilience of the existing & proposed approach is depicted in Table 5. From this table, the proposed scheme is well-equipped and is pruned against well-known attacks specifically those carried out by adversary devices to make entry in IoT or degrade its overall operation by sending false messages. The proposed system has been thoroughly checked against these attacks using a simulation environment, i.e., OMNET++, where every possible scenario related to the requesting process of a legitimate device to that of intruders or adversaries. Simulation results have confirmed that the proposed scheme is pruned against these well-known intruder attacks. The proposed approach is pruned against these attacks due to its two layers of security, i.e., MAC-based authentication and AES-128-enabled encryption that is based on a secret shared key.

Table 5: Security analysis: in terms of various attacks

Security metrics	Existing approaches					Proposed approach
	Liu et al. [22]	Hasan et al. [27]	Makhalouf et al. [26]	Gupta et al. [25]	Abdelshafy et al. [24]	
Client impersonate	√	√	√	√	√	√
Anonymity	√	√	√	√	√	√
Un-traceability	√	√	√	√	√	√
Device impersonation	√	√	√	√	√	√
Replay attack	√	√	√	√	X	√
Edge or server impersonate	X	√	X	X	√	√
Eaves-dropping	X	X	√	√	X	√
Off-line guessing	√	√	√	√	X	√
Backward and forward	√	X	√	√	√	√
Man-in-the-middle	√	X	X	√	X	√

6 Conclusion

In Internet of Things (IoT) networking infrastructures, the authenticity of communication modules, i.e., devices and servers, is mandatory to ensure the integrity of transmitted data and the privacy of source and destination devices, particularly in the presence of adversary modules. In this paper, we have developed a sophisticated, yet lightweight authentication approach designed specifically for resource-limited devices. This approach can differentiate legitimate device Ci from the adversary(s) in the IoT. The proposed approach is a hybrid of MAC and 128-bit AES to ensure that every communication session, along with handshaking, is secured against outside access in any way. Secondly, the device's mobility, i.e., moving from one position to another, is supported, and mobile devices do not require additional information for this purpose. The proposed lightweight approach is resilient against nearly all adversary attacks, especially those applicable to the IoT environmental infrastructure. Finally, the contributions of the proposed authentication scheme are verified through a sophisticated analysis of the simulation results.

Acknowledgement: Not applicable.

Funding Statement: Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2023R136), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Author Contributions: The authors confirm their contribution to the paper as follows: study conception and design: Shaha Al-Otaibi, Rahim Khan; data collection: Shaha Al-Otaibi; analysis and interpretation of results: Rahim Khan, Aftab Ahmed Khan, Jehad Ali; draft manuscript preparation: Shaha Al-Otaibi, Rahim Khan, Jehad Ali. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] A. Mirsaraei, A. Barati, and H. Barati, "A secure three-factor authentication scheme for IoT environment," *J. Parallel. Distr. Com.*, vol. 169, no. 1, pp. 87–105, 2022. doi: 10.1016/j.jpdc.2022.06.011.

[2] J. Zheng, L. Zhang, Y. Feng, and Z. Wu, "Blockchain-based key management and authentication scheme for IoT networks with chaotic scrambling," *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 1, pp. 178–188, 2022. doi: 10.1109/TNSE.2022.3205913.

[3] B. Gong, G. Zheng, M. Waqas, S. Tu, and S. Chen, "LCDMA: Lightweight cross-domain mutual identity authentication scheme for the Internet of Things," *IEEE Internet Things J.*, vol. 10, no. 14, pp. 12590–12602, 2023. doi: 10.1109/JIOT.2023.3252051.

[4] J. Hajny, P. Dzurenda, R. C. Marques, and L. Malina, "Cryptographic protocols for confidentiality, authenticity and privacy on constrained devices," in *IEEE. Int. Cong. Ultra-Modern Telecom. Control Sys. Works.*, Brno, Czech Republic, 2020, pp. 87–92.

[5] A. K. Das, M. Wazid, N. Kumar, M. K. Khan, K. K. R. Choo and Y. Park, "Design of secure and lightweight authentication protocol for wearable devices environment," *IEEE J. Biomed. Health.*, vol. 22, no. 4, pp. 1310–1322, 2017. doi: 10.1109/JBHI.2017.2753464.

[6] L. Xu, L. Chen, Z. Gao, X. Fan, T. Suh and W. Shi, "DIoTA: Decentralized-ledger-based framework for data authenticity protection in IoT systems," *IEEE Network*, vol. 34, no. 1, pp. 38–46, 2020. doi: 10.1109/MNET.001.1900136.

[7] M. A. Jan *et al.*, "Lightweight mutual authentication and privacy-preservation scheme for intelligent wearable devices in industrial-CPS," *IEEE Trans. Ind. Inform.*, vol. 17, no. 8, pp. 5829–5839, 2020. doi: 10.1109/TII.2020.3043802.

[8] J. Sun, F. Khan, J. Li, M. D. Alshehri, R. Alturki and M. Wedyan, "Mutual authentication scheme for the device-to-server communication in the Internet of medical things," *IEEE Internet Things J.*, vol. 8, no. 21, pp. 15663–15671, 2021. doi: 10.1109/JIOT.2021.3078702.

[9] S. S. Sahoo, S. Mohanty, K. S. Sahoo, M. Daneshmand, and A. H. Gandomi, "A three factor based authentication scheme of 5G wireless sensor networks for IoT system," *IEEE Internet Things J.*, vol. 10, no. 17, pp. 15087–15099, 2023. doi: 10.1109/JIOT.2023.3264565.

[10] R. Amin and G. P. Biswas, "A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks," *Ad Hoc Netw.*, vol. 36, no. 4, pp. 58–80, 2016. doi: 10.1016/j.adhoc.2015.05.020.

[11] S. R. Moosavi *et al.*, "End-to-end security scheme for mobility enabled healthcare internet of things," *Future. Gener. Comp. Sys.*, vol. 64, no. 4, pp. 108–124, 2016. doi: 10.1016/j.future.2016.02.020.

[12] B. Mbarek, M. Ge, and T. Pitner, "An efficient mutual authentication scheme for internet of things," *Int. Things*, vol. 9, no. 1, pp. 100160, 2020. doi: 10.1016/j.iot.2020.100160.

[13] P. Kumar and L. Chouhan, "A privacy and session key based authentication scheme for medical IoT networks," *Comput. Commun.*, vol. 166, no. 4, pp. 154–164, 2021. doi: 10.1016/j.comcom.2020.11.017.

[14] S. Amanlou, M. K. Hasan, and K. A. A. Bakar, "Lightweight and secure authentication scheme for IoT network based on publish-subscribe fog computing model," *Comput. Netw.*, vol. 199, pp. 108465, 2021. doi: 10.1016/j.comnet.2021.108465.
[15] M. Masud, G. S. Gaba, K. Choudhary, M. S. Hossain, M. F. Alhamid and G. Muhammad, "Lightweight and anonymity-preserving user authentication scheme for IoT-based healthcare," *IEEE Internet Things J.*, vol. 9, no. 4, pp. 2649–2656, 2021. doi: 10.1109/JIOT.2021.3080461.
[16] P. Kumar and L. Chouhan, "A secure authentication scheme for IoT application in smart home," *Peer. Peer. Netw. Appl.*, vol. 14, no. 1, pp. 420–438, 2021. doi: 10.1007/s12083-020-00973-8.
[17] V. Kumar, N. Malik, J. Singla, N. Z. Jhanjhi, F. Amsaad and A. Razaque, "Light weight authentication scheme for smart home IoT devices," *Cryptography*, vol. 6, no. 3, pp. 37, 2022.
[18] S. G. Oliver and T. Purusothaman, "Lightweight and secure mutual authentication scheme for IoT devices using CoAP Protocol," *Comput. Syst. Sci. Eng.*, vol. 41, no. 2, 2022. doi: 10.32604/csse.2022.020888.
[19] G. Fragkos, C. Minwalla, J. Plusquellic, and E. E. Tsiropoulou, "Artificially intelligent electronic money," *IEEE Consum. Electron. Mag.*, vol. 10, no. 4, pp. 81–89, 2020.
[20] R. Khan, J. Teo, M. A. Jan, S. Verma, R. Alturki and A. Ghani, "A trustworthy, reliable, and lightweight privacy and data integrity approach for the Internet of Things," *IEEE Trans. Ind. Inform.*, vol. 19, no. 1, pp. 511–518, 2022.
[21] M. F. Aziz, A. N. Khan, J. Shuja, I. A. Khan, F. G. Khan and A. U. R. Khan, "A lightweight and compromise-resilient authentication scheme for IoTs," *Trans. Emerg. Telecommun. Tech.*, vol. 33, no. 3, pp. e3813, 2022.
[22] Y. Liu, M. Dong, K. Ota, and A. Liu, "ActiveTrust: Secure and trustable routing in wireless sensor networks," *IEEE Trans. Inf. Foren. Sec.*, vol. 11, no. 9, pp. 2013–2027, 2016. doi: 10.1109/TIFS.2016.2570740.
[23] P. Gope and T. Hwang, "A realistic lightweight anonymous authentication protocol for securing real-time application data access in wireless sensor networks," *IEEE Trans. Ind. Electron*, vol. 63, no. 11, pp. 7124–7132, 2016. doi: 10.1109/TIE.2016.2585081.
[24] M. A. Abdelshafy and P. J. King, "AODV and SAODV under attack: Performance comparison," in *Ad-hoc, Mobile, and Wireless Networks*, Benidorm, Spain, Springer International Publishing, 2014, pp. 318–331.
[25] A. Gupta, M. Tripathi, T. J. Shaikh, and A. Sharma, "A lightweight anonymous user authentication and key establishment scheme for wearable devices," *Comput. Netw.*, vol. 149, no. 1, pp. 29–42, 2019. doi: 10.1016/j.comnet.2018.11.021.
[26] A. Meddeb Makhlouf and M. Guizani, "SE-AOMDV: Secure and efficient AOMDV routing protocol for vehicular communications," *Int. J. Inf. Secur.*, vol. 18, no. 5, pp. 665–676, 2019. doi: 10.1007/s10207-019-00436-z.
[27] M. R. Hasan, Y. Zhao, Y. Luo, G. Wang, and R. M. Winter, "An effective AODV-based flooding detection and prevention for smart meter network," *Comput. Netw.*, vol. 129, no. 4, pp. 454–460, 2018. doi: 10.1016/j.procs.2018.03.024.
[28] X. Li, M. H. Ibrahim, S. Kumari, A. K. Sangaiah, V. Gupta and K. K. R. Choo, "Anonymous mutual authentication and key agreement scheme for wearable sensors in wireless body area networks," *Comput. Netw.*, vol. 129, pp. 429–443, 2017. doi: 10.1016/j.comnet.2017.03.013.