IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 24, 2022, accepted May 8, 2022, date of publication May 12, 2022, date of current version May 20, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3174558

LMAS-SHS: A Lightweight Mutual Authentication Scheme for Smart Home Surveillance

SAEED ULLAH JAN[1,2], IRSHAD AHMED ABBASI[3], (Member, IEEE), AND MOHAMMED A. ALQARNI[4]

[1] Department of Computer Science & IT, University of Malakand, Chakdara, Khyber Pakhtunkhwa 18800, Pakistan
[2] Department of Computer Science, Government Degree College Wari (Dir Upper), Wari, Dir Upper 18200, Pakistan
[3] Faculty of Science & Arts Belqarn, Department of Computer Science, University of Bisha, Sabt Al-Alaya 61985, Saudi Arabia
[4] Department of Software Engineering, College of Computer Science and Engineering, University of Jeddah, Jeddah 23890, Saudi Arabia

Corresponding author: Irshad Ahmed Abbasi (aabasy@ub.edu.sa)

This work was supported by the University of Bisha, Saudi Arabia.

ABSTRACT The network-enabled smart objects are evolving everywhere in the modern era to solve numerous problems like real-world data collection from the environment, communication, analysis, and security. However, these objects (Internet-of-Things), in combination with low latency networks, are still not qualified for complex tasks and do not deliver efficient services due to the restriction of access and lack of secure authentication protocol. Because the data is collected by the embedded sensors inside the smart object in a real-time manner from the environment and communicated to the destination Centre (server) for intelligent decisions, are vulnerable to numerous threats, attention is required for the security and needs to be secure, as it transmits via an open network channel. This security issue can only be handled by designing a flawless, lightweight, and robust mutual authentication scheme. To do so, we have proposed a mutual authentication scheme using a simple hash cryptographic function, Elliptic Curve Cryptographic (ECC) technique, and XOR operations. The proposed scheme is lightweight, efficient, and effective in performance while offering secure transmission sessions among all the participants. The security of the proposed mechanism has been formally tested using GNY (Gong-Needham-Yahalon) logic, ProVerif2.03, and informally using propositions and realistic discussions. By comparing it with many of the existing authentication protocols, it has been demonstrated that our scheme is lightweight in terms of computation and communication metrics.

INDEX TERMS Authentication, cryptography, security, encryption, curve, gateway, verification, GNY.

I. INTRODUCTION

The smart object is placed in a building, bridge, at home etc. and then connected to the internet for providing services to users at any time and from any location. This technology benefits various application sectors, including healthcare systems [1], transportation surveillance, infrastructure inspection, and home monitoring. In 2020 the estimated IoT device industry was round about 25 billion USD, while the prediction for 2025 is approximately 6 trillion USD [2]. In the smart home scenario, the user can enjoy a high level of convenience using IoT devices, such as if the user desires to turn on/off lights, open/close doors, increase/decrease temperature, check surveillance, etc., user can easily control these smart objects (IoT) from any comfort zone using a portable device. However, smart objects are inefficient in terms of computing storage and battery consumption. A robust authentication system is difficult to establish since embedded devices have limited storage. In order to deploy smart objects for smart home monitoring, vigorous authentication and lightweight security system are required, and acceptable carefulness for user ease in managing the system [3] is needed. So that malicious entities cannot gain illegal access to the data sent towards the smart object (IoT) [3]. As many homes are now-a-days internet-connected and visible to the public and malicious actions could easily compromise a user's privacy. An intruder, for example, could eavesdrop on data exchanged between a user's mobile and a smart object(s), and by collecting data repeatedly, they could then estimate when the homeowner wakes up, leaves for work, sleeps, and even travels. An attacker may prepare more serious attacks based on the eavesdropped data, such as burglary, kidnapping, and theft [4]. Majority of smart homes' authentication techniques described in the literature are insecure against various attacks, including insider attacks, gateway node and smart object impersonation

The associate editor coordinating the review of this manuscript and approving it for publication was Xiangxue Li.

attacks, smart card theft attacks, and denial-of-service (DoS) attacks.

The lifestyle of a layman has changed due to the recent development of high-speed internet and the increased use of IoT in homes, cities, healthcare, industries, and security and rescue operations. IoT devices communicate with each other and with a centralized control system for different functions remotely. However, the security and privacy of IoT devices are still a big issue for researchers because of their heterogeneous nature. In this regard, different researchers have designed numerous authentication mechanisms using diverse cryptographic techniques. So far, Zhang *et al.* [5] proposed a simple hash and XOR-based authentication and key agreement scheme (AKA) for Internet-of-Drones (IoD) in which Unmanned Aerial Vehicle (UAV) has been used for search and rescue operations. However, their scheme is vulnerable as it suffers from physical capture, side-channel, and time synchronization attacks and has design flaws. Jan *et al.* [6], [7] proposed Hash Message Authentication Code Secure Hash Algorithm (HMACSHA1) and Public Key Infrastructure (PKI)-based authentication schemes for securing IoD in which drones can be used for different activities like wildlife surveillance, pipe-line inspection, sidewalk monitoring and agricultural spraying. However, after analysis, it has been demonstrated that their protocols' computation and communication metrics are still not up to the mark, requiring more effort to make them lightweight.

Won *et al.* [8] presented three certificateless cryptographic schemes for real-world environmental monitoring using smart objects placed at different buildings, bridges and stations. They claimed that their first efficient certificateless signcryption tag key encapsulation mechanism (eCLSC-TKEM) is operational when data transmission is performed between smart objects and drones. Their [8] second certificateless multi-recipient encryption scheme (CL-MRES) is applicable when services are delivered from a drone to many smart objects. Their [8] last certificateless data aggregation (CLDA) protocol's functioning when broadcasting is performed among many small objects towards a drone. However, these schemes are difficult to implement due to aggregate data verification instead of one-to-one. Wazid *et al.* [9] designed an Elliptic Curve Cryptographic (ECC) based authentication scheme for IoD deployment in the civilian domain, but their plan could also not be implemented practically due to less power in the smart objects and increased processing of data.

Furthermore, some available security protocols are weak against desynchronization and stolen verifier attacks. For example, in 2020, Hong *et al.* [10] demonstrated a security mechanism for reconnaissance and attacking drones equipped with many smart objects and are deployed in clusters for real-world environment monitoring. The [10] claimed that the airborne control and command platform (AC2P) establishes information broadcasting for reconnaissance and attacking drones, communicating real-time information collected from the environment to the centralized control system for an intelligent decision. After an extensive analysis of their [10] protocol, it has shown that their security mechanism is suffering from desynchronization, stolen-verifier, and privileged insider attacks. An improved protocol suite presented by Jan *et al.* [11] for IoD deployment military drones using the concept of pairing cryptography and identity authentication. They [11] have significantly tackled the weaknesses of [10] and claimed that a drone alone could not perform a complex tactical task; drones must be operationalized in many clusters subject to collaboration and coordination among them. However, the performance analysis of [11] still needs more effort for modification. Using the notion of symmetric-key cryptosystems, [12] introduced a lightweight authentication protocol for vehicle-to-infrastructure communication (V2I) in which successful authentication with the trusted third party (TA), the system distributed a secret key among the car and the roadside unit (RSU) or many smart objects to allow them to communicate with the server for real-time traffic monitoring. Although their [12] strategy used lightweight operations like hash and XOR function but is still insufficient for the restricted resources to make the system credible. Their scheme [12] also wastes a significant amount of storage space when storing the secret internal parameters among all the participants. Another disadvantage of their [12] method is that it does not accomplish reachability security features.

Similarly, the smart objects are also used to collect the most relevant environmental and climatic data for agricultural land to determine the amount of water pumped to the crop. Humidity, rain, soil moisture, soil pH, light, and temperature sensors are among the sensors used [13] for operating smart objects in these activities. Then, the system and analyzer servers transmit and analyze the said environmental and climatic data and forward the processed results to the responsible agriculture expert, who can then issue instantaneous commands to the water pumping actuators. Also, by using various machine learning methods, the system analyzer server can assess the arriving environmental and meteorological data to estimate the required water level. Again, most existing smart objects have used the Internet of Things (IoT) and Wireless Sensor Network (WSN) technologies in agriculture development to establish communication channels amongst stakeholders for onward decisions regarding productive crop output [14]. The said sensitive activities require a fast CPU and large memory space in the smart object/sensors, which is impossible for such a tiny device. A security mechanism with less computation/communication costs and robust security is the only solution for achieving the earlier goals related to agricultural land.

Finally, as stated above, most of the security protocols available in the literature are either unprotected in maintaining traceability and user anonymity or having high costs due to modular exponentiation. An insider threat is noted in these schemes in which an attacker uses the power analysis technique to steal the users' confidential credentials, such as identification, private certificates, and password. While in an impersonation attack, the intruder may produce genuine messages and send them to the appropriate smart object, causing the messages to be treated as legitimate but later on show severe damage to the whole system. An attacker could also utilize the extracted information from a stolen smart object to deduce the user's secret credentials and later use it for malicious deeds. Therefore, LMAS-SHS has been presented here in this article to address these drawbacks. The key contributions are as under:

1. LMAS-SHS is an ECC-based lightweight security mechanism, which requires less power consumption due to reduced computation costs due to compressed message size.
2. LMAS-SHS is secure; the security of LMAS-SHS has been scrutinized on two methods:
   i. GNY Logic is used for checking the hash values and the security of random numbers exchanged among participants.
   ii. A programming verification toolkit, ProVerif2.03, has been used for checking the session key secrecy, confidentiality, and reachability.
3. LMAS-SHS is lightweight, efficient, and effective in performance while offering secure transmission sessions among all the participants.
4. A pragmatic illustration for different attacks shows that LMAS-SHS resists all known attacks.

A. NETWORK MODEL

The proposed model for the network consisted of three main entities, i.e., mobile user (M), gateway (trusted entity), and smart object (having low power capability and less memory). The smart object sends real-time information to the gateway node. The mobile user is connected with the gateway node to disseminate real-time information fusion to the gateway node, as shown in Fig, 1. It is worth mentioning that the smart home must be equipped with smart objects for measuring different conditions like proximity, humidity, temperature, and the entire IoT installed devices like air conditions, fans, doors, locks, refrigerators, etc. Secondly, the service of the gateway node must be in control of home appliances on system architecture. The mobile user can use the managing service to regulate the operations of smart actuators connected to home equipment like lamps and fans. Finally, there must be a robust mutual authentication and lightweight cross-verification protocol that can access the entire entities having authorized people's identification characteristics.

B. ADVERSARY MODEL

Modelling the role of attackers is essential in cyber defense since it helps to guarantee that security assessments are scientifically sound, especially for conceptual contributions that are difficult to test or where comprehensive testing is impossible. In a computer or networked system, an adversary model is a formalization of an attacker. Depending on how extensive this formalization is, the opponent might be an algorithm or a collection of affirmations about skills and discretions. This umbrella confines a variety of techniques in many domains of computer security. Therefore, keeping in view the adversary model, an adversary interacts with our smart home security architecture by representing themselves as a malicious user through gateway node in the following manner.

1. An adversary may extract stored data from gateway memory and use it to verify the secret credentials of a legitimate user for malicious deeds.
2. An adversary may alter, erase, upgrade, corrupt, or insert false information in the transmitted data over a public network channel.
3. An Adversary may replay, alter, or erase beneficial information exchanged between participants over a private channel.
4. An adversary may achieve the goal of entering the internal sensitive credential from a stolen smart object or mobile device.
5. An adversary might shape the memory of a stolen or misplaced smart object using reverse engineering approaches or key tags in offline mode, but not both simultaneously.

Therefore, to make the system efficient and effective, a cryptographic-based protocol is mandatory for adequately achieving integrity, confidentiality, authentication, and non-repudiation, ensuring perfect forward secrecy in unfavourable channels between participants. Regardless of who is involved, all legal parties in a session must trust one another to meet specific information security-related goals, which are the aim of this research.

C. ELLIPTIC CURVE CRYPTOGRAPHY (ECC)

Mathematicians and cryptographers introduced ECC for cryptosystem (a public key cryptographic method). It is lightweight based on an algebraic pattern of a curve over a finite field. It can deliver better security, faster computation, and network broadcasting. ECC is used for key controlling and authentication and can be defined in the following equation:

$$y^2 = x^3 + ax + b \tag{1}$$

i. Suppose P and Q are two points in eq: (1), then its addition can be represented as $P + Q = R$ whereas $P \neq Q$. In the curve, the lines via P and Q intersect at R.
ii. Suppose $Q = -P$, then $P + Q = P + (-P) = P - P = 0$, which means P and $-P$ interest the cure at Q, called the point of infinity.
iii. Suppose P is itself taken like $P + P = 2P = Q$, which means P intersects $-Q$ and is reflected over the x-axis at Q.
iv. Suppose a point is added to the curve is k means k. $P = P + P + P \ldots P$ (k times) $=$ whereas $k \in Z_p^*$ in the cyclic group G.
v. ECC is smaller, as RSA occupies 1024 bits of memory, while ECC is just 160 bits of space.

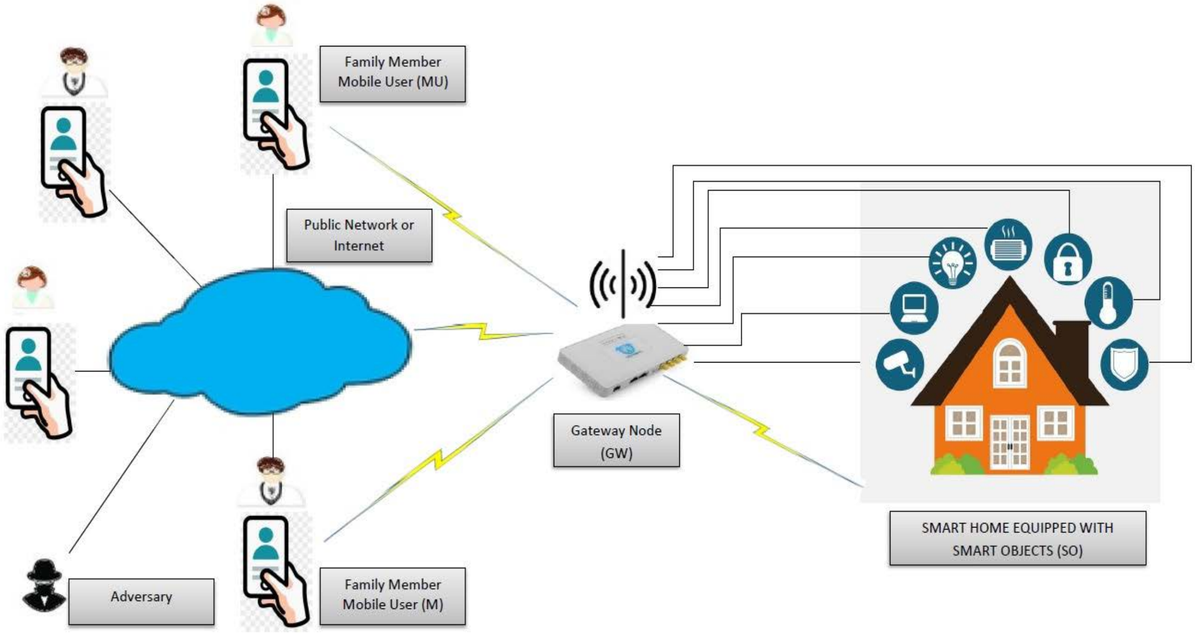

Family Member Mobile User (MU)
Public Network or Internet
Gateway Node (GW)
SMART HOME EQUIPPED WITH SMART OBJECTS (SO)
Adversary
Family Member Mobile User (M)

FIGURE 1. System model.

TABLE 1. Notations and their description.

Notation	Description	Notation	Description
M	Mobile User	G	Gateway Node
ID_M	Mobile Identity	ID_G	Gateway Identity
K_M	Mobile Public Key	K_G	Gateway Public Key
S	secret key	R	Random number
N_M	Mobile's Nonce	N_G	Gateway Nonce
P	Curve Point	F_q	Finite Field
SO	Smart Object	K_S	session Key
ID_{SO}	Smart-Object Identity	K_{SO}	Smart-Object Nonce
K_{SO}	Smart-Object Public Key	p	prime number

vi. An ECC-based key solution is almost impossible. It requires steps to be solved, whereas P means cardinality of the curve (a large prime number).

vii. Require less power consumption due to smaller computation and compressed message size; therefore, ECC is recommended for the resource-constrained environment.

II. PROPOSED SCHEME

We propose an ECC-based, lightweight, and secure mutual authentication scheme for smart home surveillance called LMAS-SHS. The LMAS-SHS consists of i) setup phase, ii) registration phase, and iii) mutual authentication phase. These phases are described under the following headings, while the different notations used for designing LMAS-SHS are shown in Table 1.

A. SETUP PHASE

The gateway node selects a curve $E(F_q)$ of equation $y^2 = x^3 + ax + b$ whereas a, b ϵ F_q. Suppose all the three participants, i.e., mobile device (M), gateway node (GW), and smart object (SO), select their secret keys s, compute public key $K_M = s.P$, $K_G = s.P$, and $K_{SO} = s.P$, whereas P means a point in the curve.

B. REGISTRATION PHASE

This phase of LMAS-SHS is completed in the following two sub-phases:

1) MOBILE (M) REGISTRATION PHASE

This sub-phase is completed in the following set of computations:

MR1: First, a unique identity is generated ID_M, select a nonce N_M, compute $PID_M = ID_M||N_M$, $M = PID_M||K_M$ and transmits (I_M, ID_M) towards the gateway node (GW) over a secure channel.

MR2: The gateway node (GW), upon receiving (I_M, ID_M) message can also generate a nonce N_G, random number R_G and compute $S_1 = N_G \oplus N_M$, $S_2 = S_1 \oplus R_G$, $S_3 = h(S_1||ID_M)$, $S_4 = h(h(S_2||ID_M)||N_M)$, $S_5 = h(ID_M||K_M||N_M)$, $S_6 = h(ID_G||K_G||N_G)$, $S_7 = R_G \oplus S_5 \oplus S_6$, $S_8 = N_G \oplus S_5 \oplus S_6$ stores $(ID_M, h(S_1||ID_M), S_2)$ and transmits $(S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, N_G)$ message back towards mobile (M) over a secure channel.

MR3: The mobile (M) confirms N_G and computes $S_1^{/} = N_G \oplus N_M$, $S_2^{/} = S_1^{/} \oplus R_G$, $S_5^{/} = h(ID_M||K_M||N_M)$, $S_6^{/} = h(ID_G||K_G||N_G)$, $S_1^{/} = N_G \oplus N_M$ and stores $(ID_M. ID_G, h(S_1^{/}||ID_G), S_2)$, as shown in module 1(a).

2) SMART OBJECT (SO) REGISTRATION PHASE

This sub-phase of LMAS-SHS competed in these steps:

MOBILE (M)	GATEWAY NODE (GW)

Selects identity ID_M, nonce N_M
Compute: $PID_M=ID_M||N_M$
$I_M=PID_M||K_M$

$\xrightarrow{I_M,\ ID_M}$

Confirms: N_M
Generate Identity ID_G, Nonce N_G
Selects random number R_G
Computes: $S_1=N_G \oplus N_M$
$S_2=S_1 \oplus R_G$
$S_3=h(S_1||ID_M)$
$S_4=h(S_1||ID_M||N_M)$
$S_5=h(ID_M||K_M||N_M)$
$S_6=h(ID_G||K_G||N_G)$
Stores $\{ID_M, h(S_1||ID_M), S_2\}$

$\xleftarrow{S_1,\ S_2,\ S_4,\ S_5,\ S_6,\ R_G}$

Confirms: R_G
Computes: $S_1^{/}=N_G \oplus N_M$
$S_2^{/}=S_1 \oplus R_G$
$S_4^{/}=h(S_2||ID_M)||N_M$
Verify $S_4^{/}?=S_4$
$S_5^{/}=h(ID_M||K_M||N_M)$
$S_6^{//}=h(ID_G||K_G||N_G)$ and stores: $\{ID_M, ID_G, h(S_1||ID_G), S_2\}$

SMART OBJECT (SO)	GATEWAY NODE (GW)

Selects identity ID_{SO}, nonce N_{SO}
Compute: $PID_{SO}=ID_{SO}||N_{SO}$
$I_{SO}=PID_{SO}||K_{SO}$

$\xrightarrow{I_{SO},\ ID_{SO}}$

Confirms: N_{SO}
Generate Identity ID_G, Nonce N_G
Selects random number R_G
Computes: $Z_1=N_G \oplus N_{SO}$
$Z_2=Z_1 \oplus R_G$ and $Z_3=h(Z_1||ID_{SO})$
$Z_4=h(Z_1||ID_{SO}||N_{SO})$
$Z_5=h(ID_{SO}||K_{SO}||N_{SO})$
$Z_6=h(ID_G||K_G||N_G)$
Stores $\{ID_{SO}, h(Z_1||ID_{SO}), Z_2\}$

$\xleftarrow{S_1,\ S_2,\ S_4,\ S_5,\ S_6,\ R_G}$

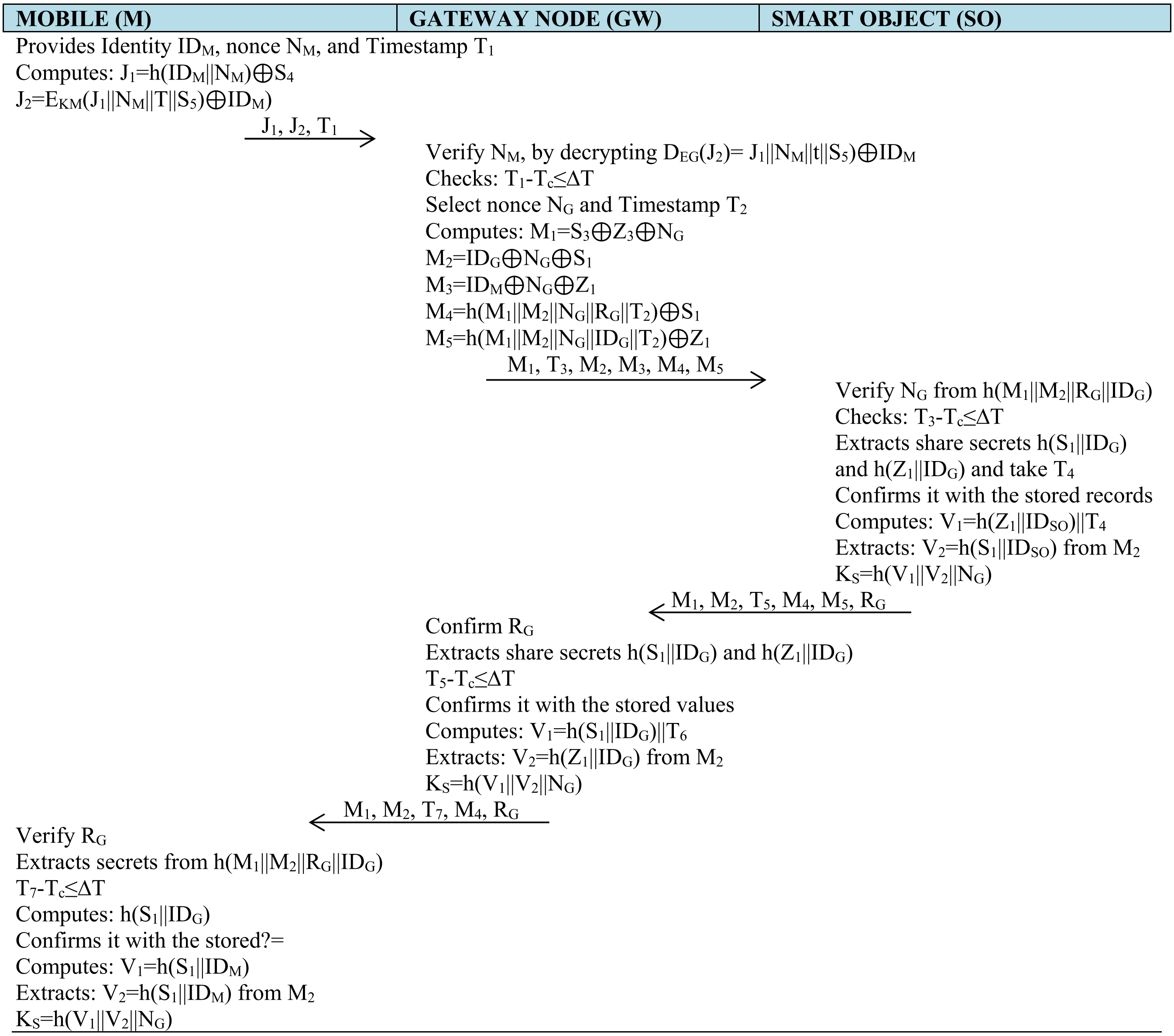
Confirms: R_G
Computes: $Z_1^{/}=N_G \oplus N_{SO}$
$Z_2^{/}=Z_1 \oplus R_G$ and $Z_4^{/}=h(Z_2||ID_{SO}||N_{SO})$
Verify $Z_4^{/}?=Z_4$
$Z_5^{/}=h(ID_{SO}||K_{SO}||N_{SO})$
$Z_6^{/}=h(ID_G||K_G||N_G)$ and Stores: $\{ID_{SO}, ID_G, h(Z_1||ID_G), Z_2\}$

MODULE 1. (A) Mobile registration phase. (B) Smart Object (SO) registration phase.

SOR1: The smart object (SO) first selects identity ID_{SO}, random nonce N_{SO}, and computes: $PID_{SO} = ID_{SO}||N_{SO}$, $I_{SO} = PID_{SO}||K_{SO}$, transmits (I_M, ID_{SO}) message towards gateway node (GW) over a secure channel.

SOR2: Upon receiving (I_M, ID_{SO}) message, the gateway node (GW) confirms N_{SO}, generates ID_G, random nonce N_G, selects random number R_G and computes: $Z_1 = N_G \oplus N_{SO}$, $Z_2 = Z_1 \oplus R_G$, $Z_3 = h(Z_1||ID_{SO})$, $Z_4 = h(h(Z_1||ID_{SO})||N_{SO})$, $Z_5 = h(ID_{SO}||K_{SO}||N_{SO})$, $Z_6 = h(ID_G||K_G||N_G)$, stores $(ID_{SO}, h(Z_1||ID_{SO}), Z_2)$ in its memory and sends $(S_1, S_2, S_4, S_5, S_6, N_G)$ towards the smart object (SO) over a secure path.

SOR3: The smart object (SO), when receiving $(S_1, S_2, S_4, S_5, S_6, N_G)$ message from the gateway (GW), first confirms N_G and computes: $S_1^{/} = N_G \oplus N_{SO}$, $Z_2^{/} = Z_1 \oplus R_G$, $Z_4^{/} = h(h(Z_2||ID_{SO})||N_{SO})$, verify $Z_4^{/}? = Z_4$, $Z_5^{/} = h(ID_{SO}||K_{SO}||N_{SO})$, $Z_6^{//} = h(ID_G||K_G||N_G)$, and finally stores $(ID_{SO}, ID_G, h(Z_1||ID_G), Z_2)$ values in the memory of smart object (SO) which will be used later in the authentication, as shown in module 1(b).



MOBILE (M)	GATEWAY NODE (GW)	SMART OBJECT (SO)

Provides Identity ID_M, nonce N_M, and Timestamp T_1
Computes: $J_1=h(ID_M||N_M)\oplus S_4$
$J_2=E_{KM}(J_1||N_M||T||S_5)\oplus ID_M)$

$\xrightarrow{J_1, J_2, T_1}$

Verify N_M, by decrypting $D_{EG}(J_2)= J_1||N_M||t||S_5)\oplus ID_M$
Checks: $T_1\text{-}T_c\leq\Delta T$
Select nonce N_G and Timestamp T_2
Computes: $M_1=S_3\oplus Z_3\oplus N_G$
$M_2=ID_G\oplus N_G\oplus S_1$
$M_3=ID_M\oplus N_G\oplus Z_1$
$M_4=h(M_1||M_2||N_G||R_G||T_2)\oplus S_1$
$M_5=h(M_1||M_2||N_G||ID_G||T_2)\oplus Z_1$

$\xrightarrow{M_1, T_3, M_2, M_3, M_4, M_5}$

Verify N_G from $h(M_1||M_2||R_G||ID_G)$
Checks: $T_3\text{-}T_c\leq\Delta T$
Extracts share secrets $h(S_1||ID_G)$
and $h(Z_1||ID_G)$ and take T_4
Confirms it with the stored records
Computes: $V_1=h(Z_1||ID_{SO})||T_4$
Extracts: $V_2=h(S_1||ID_{SO})$ from M_2
$K_S=h(V_1||V_2||N_G)$

$\xleftarrow{M_1, M_2, T_5, M_4, M_5, R_G}$

Confirm R_G
Extracts share secrets $h(S_1||ID_G)$ and $h(Z_1||ID_G)$
$T_5\text{-}T_c\leq\Delta T$
Confirms it with the stored values
Computes: $V_1=h(S_1||ID_G)||T_6$
Extracts: $V_2=h(Z_1||ID_G)$ from M_2
$K_S=h(V_1||V_2||N_G)$

$\xleftarrow{M_1, M_2, T_7, M_4, R_G}$

Verify R_G
Extracts secrets from $h(M_1||M_2||R_G||ID_G)$
$T_7\text{-}T_c\leq\Delta T$
Computes: $h(S_1||ID_G)$
Confirms it with the stored?=
Computes: $V_1=h(S_1||ID_M)$
Extracts: $V_2=h(S_1||ID_M)$ from M_2
$K_S=h(V_1||V_2||N_G)$

MODULE 2. Mutual authentication phase.

C. MUTUAL AUTHENTICATION PHASE

This phase of the scheme is completed in the following steps:

A1: The user, through a mobile device, provides identity ID_M, computes: $J_1 = ID_M||N_M$, $J_2 = J_1 \oplus K_M$ and transmits $\{S_4, S_5, Z_4, Z_5, J_1, J_2\}$, whereas S_4, S_5, Z_4, Z_5 are the stored values in registration phase.

A2: The gateway node first verifies N_M, and N_G from S_3 and Z_3, computes: $M_1 = S_3 \oplus Z_3 \oplus N_G$, $M_2 = ID_G \oplus N_G \oplus S_1$, $M_3 = ID_M \oplus N_G \oplus Z_1$, $M_4 = h\,(M_1||M_2||N_G||ID_G) \oplus S_1$, $M_4 = h(M_1||\ M_2||N_G||ID_G) \oplus Z_1$, and transmits $\{M_1, M_2, M_3, M_4\}$ towards smart object over wireless channel.

A3: Here, first verify N_G from $h(M_1||M_2||N_G||ID_G)$, and extracts share secrets $h(S_1||ID_G)$, confirms it with the stored values, computes: $V_1 = h(Z_1||ID_{SO})$ and extracts: $V_2 = h(S_1||ID_G)$ from M_2 and compute session key $K_S = h(V_1||V_2||N_G)$. Transmits the message except M_3 back towards gateway node $\{M_1, M_2, M_4\}$.

A4: There first verify N_G from $h(M_1||M_2||N_G||ID_G)$, extracts share secrets $h(S_1||ID_G)$, confirms it with the stored, computes: $V_1 = h(S_1||ID_G)$, again extracts: $V_2 = h(Z_1||ID_{SO})$ from M_2 and computes session key $K_S = h(V_1||V_2||N_G)$. Also, transmits the message except M_3 back towards gateway node $\{M_1, M_2, M_4\}$,

A5: The user upon receiving $\{M_1, M_2, M_4\}$ message, first verify N_G, extracts secrets from $h(M_1||M_2||N_G||ID_G)$, computes: $h(S_1||ID_G)$, confirms it with the stored record, computes: $V_1 = h(S_1||ID_M)$, extracts: $V_2 = h(S_1||ID_G)$ from M_2 and calculates session key $K_S = h(V_1||V_2||N_G)$ as shown in module 2.

III. SECURITY ANALYSIS

This section analyzes the cryptographic protocol's trust, freshness, and robustness and designates a protocol's correctness. Also, it tells the readers why widespread authentication protocol attacks occur, and then it addresses the robustness based on trustworthiness and freshness. Therefore, keeping in view the goals mentioned above, we will scrutinize the security of LMAS-SHS formally using:

a) Gong-Needham-Yahalon called GNY logic [15], b) programming verification software toolkit ProVerif2.03; and informally using sensible explanation and arguments.

A. GNY LOGIC ANALYSIS

This section analyses our proposed scheme using GNY logic [15]. First, we introduce statements and formulae used in GNY logic; after that, we define the goals and assumptions of our proposed protocol. In the end, we verify the security of our proposed scheme using GNY logic [15].

1) GNY LOGIC FORMULAS AND STATEMENTS

The formulae used in GNY logic are as follows

a) (W, N): the conjunction of two formulae W and N
b) $\{X\}_K$ and $\{K\}_K^{-1}$ encryption/decryption with key K
c) h(W): A one-way function of W
d) *W: W is not originated here

Here the W and N are the range over formulae while X and Y are the principals. The statements used in LMAS-SHS are as follow.

a) $X \triangleleft W$: X is told formula W
b) $P \ni W$: W possesses formula W
c) $X| \backsim W$: W once conveyed formula W
d) $X| \equiv \#(W)$: W believes that W is fresh
e) $P | \equiv \phi(W)$: W believes that W is recognizable
f) $X| \equiv W \xleftrightarrow{S_K} Y$: P believes that S_K is a suitable secret key for X and Y
g) $X| \Rightarrow W$: P has jurisdiction over X
h) $X \triangleleft^* W$: X is told that formula X which did not convey previously in the current run.

To apt the GNY logic with the LMAS-SHS, we make several notations' changes as under:

(1) $M \rightarrow GW: (\{J_1, J_2, T_1\})$
(2) $GW \rightarrow SO: (\{M_1, T_3, M_2, M_3, M_4, M_5\})$
(3) $SO \rightarrow GW: (\{M_1, M_2, T_5, M_4, M_5, R_G\})$
(4) $GW \rightarrow M: (\{M_1, M_2, T_7, M_4, R_G\})$

2) GNY LOGIC ASSUMPTION FOR LMAS-SHS

A1: $M \ni \#(N_M)$ A10: $GW \ni \#(N_M)$ A19: $SO \ni \#(N_M)$
A2: $M \ni \#(t_1)$ A11: $GW \ni \#(t_1)$ A20: $SO \ni \#(t_1)$
A3: $M \ni \#(N_G)$ A12: $GW \ni \#(N_G)$ A21: $SO \ni \#(N_G$
A4: $M \ni \#(T_2)$ A13: $GW \ni \#(T_2)$ A22: $SO \ni \#(T_2)$
A5: $M \ni \#(N_{SO})$ A14: $GW \ni \#(N_{SO})$ A23: $SO \ni \#(N_{SO})$
A6: $M \ni \#(R_G)$ A15: $GW \ni \#(R_G)$ A24: $SO \ni \#(R_G)$
A7: $M \ni ID_M$ A16: $GW \ni ID_M$ A25: $SO \ni ID_M$
A8: $M \ni ID_G$ A17: $GW \ni ID_G$ A26: $SO \ni ID_G$
A9: $M \ni ID_{SO}$ A18: $GW \ni ID_{SO}$ A27: $SO \ni ID_S$

It means the nonce generated by M, said that N_M is fresh ($\#N_M$) and possesses K_{MN}, S_4, T_1, and ID_M. Similarly, the N_G and R_G are fresh ($\#R_G$), ($\#N_G$) and possesses S_1, T_2, ID_M and ID_G. Also, the K_S generated on smart object (SO) side is fresh ($\#K_S$) and possesses ID_{SO}, T_4, Z_1, ID_G, and M_2.

$$M \ni N_M, M \ni (\#N_M), M \ni K_{KM}, M \ni S_4,$$
$$M \ni T_1, M \ni ID_M$$
$$GW \ni T_2, GW \ni S_1, GW \ni ID_M, GW \ni N_G,$$
$$GW \ni R_G, GW \ni ID_G$$
$$SO \ni ID_{SO}, SO \ni Z_1, SO \ni T_4, SO \ni ID_G, SO \ni M_2$$

3) GNY LOGIC ANALYSIS FOR LMAS-SHS

We changed the notation of LMAS-SHS to fill in GNY logic.

a) $M \rightarrow GW: \{h(ID_M||N_M) \oplus S_4\}$
$M \rightarrow \{(J_1||N_M||t||S_5) \oplus ID_M)\}$
$M \rightarrow \{T_1\}$
b) $GW \rightarrow SO: \{(M_1, T_3, M_2, M_3, M_4, M_5)\}$
c) $SO \rightarrow GW: \{S_3 \oplus Z_3 \oplus N_G\}$
$SO \rightarrow \{ID_G \oplus N_G \oplus S_1\}$
$SO \rightarrow \{h(M_1||M_2||N_G||ID_G||T_2) \oplus S_1\}$
$SO \rightarrow \{h(M_1||M_2||N_G||ID_G||T_2) \oplus Z_1\}$
$SO \rightarrow \{N_G\}$
d) $GW \rightarrow M: \{S_3 \oplus Z_3 \oplus N_G\}$
$GW \rightarrow \{ID_G \oplus N_G \oplus S_1\}$
$GW \rightarrow \{h(M_1||M_2||N_G||ID_G||T_2) \oplus S_1\}$
$GW \rightarrow \{N_G\}$

The exchange among the participants in the protocol has completely performed secretly like M believes that M → GW exchanges $ID_M \| N_M) \oplus S_4$ and J_1, J_2, T_1 secretly. In the same way the exchange of credentials between GW → SO are exchanged much secretly like $(M_1 \| M_2 \| N_G \| R_G) \oplus S_1$, $(M_1 \| M_2 \| N_G \| R_G) \oplus Z_1$, $M_1 \| M_2 \| N_G \| R_G \| T_2) \oplus ID_M$, and $M_1, M_2, M_3, M_4, M_5, T_2$. Among SO → GW and GW → M, the credentials exchanged are $M_1, M_2, M_4, M_5, T_5, R_G$ and $M_1, M_2, M_4, M_7, T_7, R_G$ performed in a secure manner.

$$M| \equiv M \xleftrightarrow{(ID_M \| N_M) \oplus S_4} GW$$
$$M| \equiv M \xleftrightarrow{J_1, J_2, T_1} GW$$
$$GW| \equiv GW \xleftrightarrow{(M_1 \| M_2 \| N_G \| R_G) \oplus S_1} SO$$
$$GW| \equiv GW \xleftrightarrow{(M_1 \| M_2 \| N_G \| R_G) \oplus Z_1} SO$$
$$GW| \equiv GW \xleftrightarrow{(M_1 \| M_2 \| N_G \| R_G \| T_2) \oplus ID_M} SO$$
$$GW| \equiv GW \xleftrightarrow{M_1, M_2, M_3, M_4, M_5, T_2} SO$$
$$SO| \equiv SO \xleftrightarrow{M_1, M_2, M_4, M_5, T_5, R_G} GW$$
$$GW| \equiv GW \xleftrightarrow{M_1, M_2, M_4, M_7, T_7, R_G} M$$

4) GNY LOGIC GOALS FOR LMAS-SHS

1) Goal 1: $GW| \equiv N_M$
2) Goal 2: $SO| \equiv N_G$

3) Goal 3: $\mathrm{GW}| \equiv \mathrm{N_G}$
4) Goal 4: $\mathrm{M}| \equiv \mathrm{N_G}$

According to assumptions A1 and A3, we get

$$\frac{M \ni N_G, M \ni N_M}{M \ni S_1}$$

According to assumptions A7 and A1, we get

$$\frac{M \ni S_1, M \ni ID_M, M \ni N_M}{M \ni S_4}$$

According to assumptions A7 and A1, we get

$$\frac{M \ni ID_M, M \ni N_M, M \ni S_4}{M \ni J_1}$$

According to assumptions A16, A10, and GNY postulates, T1 and P1

$$\frac{\mathrm{GW} \triangleleft J_1}{\mathrm{GW} \ni J_1}$$

According to assumptions A7, A1, A10, we obtain

$$\frac{\mathrm{M} \ni J_1, \mathrm{M} \ni N_M, \mathrm{M} \ni S_5, \mathrm{M} \ni t}{\mathrm{M} \ni J_2}$$

According to assumptions A10, A11, and GNY postulates, T1 and P1

$$\frac{\mathrm{GW} \triangleleft J_2}{\mathrm{GW} \ni J_2}$$

According to assumptions A11, A11, A16, and GNY postulates, T1 and P1

$$\frac{\mathrm{GW} \ni J_1, \mathrm{GW} \ni N_M, \mathrm{GW} \ni t, \mathrm{GW} \ni S_5, \mathrm{GW} \ni ID_M}{\mathrm{GW} \ni J_2}$$

The user sees the identity of gateway along with the message communicated with key $\mathrm{K_M}$, and then gateway believes its identity and message received.

$$\frac{\mathrm{M} \triangleleft ID_G, M \triangleleft \{\mathrm{J_1, J_2, T_1}\}_{K_M}}{\mathrm{GW} \ni ID_G, GW \ni \{\mathrm{J_1, J_2, T_1}\}_{K_M}}$$

The GW, now possesses $\mathrm{ID_G}$ and $\{\mathrm{J_1, J_2, T_1}\}_{K_M}$, so, we can represent it by the following form

$$\frac{\mathrm{GW} \ni ID_G, GW \ni R_G}{\mathrm{GW} \ni ID_G, GW \ni (ID_G||R_G)}$$

In order to achieve Goal 1, the GW recognizes $\mathrm{J_2}$ and applies R1

The *SO* sees part of the message and computes. According to assumptions A19 and A21, we obtain

$$\frac{SO \ni N_G, SO \ni N_M}{SO \ni S_1}$$

According to assumptions, A25 and GNY postulate T1 and P1

$$\frac{SO \ni S_1, SO \ni ID_M}{SO \ni S_3}$$

According to assumptions A21 and A23, we obtain

$$\frac{SO \ni N_G, SO \ni N_{SO}}{SO \ni Z_1}$$

According to A25 and GNY postulates, T1 and P1

$$\frac{SO \ni Z_1, SO \ni ID_{SO}}{SO \ni Z_2}$$

According to assumption A25 and GNY T1 and P1, we get

$$\frac{SO \ni S_3, SO \ni Z_3, SO \ni N_G}{, SO \ni M_1}$$

According to assumptions A26, A21, and GNY T1 and P1, we obtain

$$\frac{SO \ni ID_G, SO \ni N_G, SO \ni S_1}{SO \ni M_2}$$

$$\frac{\mathrm{SO} \triangleleft M_2}{\mathrm{SO} \ni M_2}$$

If the smart object (SO) sees the identity of mobile along with the message communicated, then smart object (SO) can definitely believe its own identity and the message broadcasted.

$$\frac{\mathrm{SO} \triangleleft ID_M, SO \triangleleft \{\mathrm{M_1, M_2, M_4, M_7, T_7, R}_G\}_{R_G}}{\mathrm{SO} \ni ID_{SO}, SO \ni \{\mathrm{M_1, M_2, M_4, M_7, T_7, R}_G\}_{R_G}}$$

The smart object possesses $\mathrm{ID_{SO}}$ and $\mathrm{M_1, M_2, M_4, M_7, T_7, R}_G$, so, we can represent it by the following form:

$$\frac{\mathrm{GW} \ni ID_{SO}, GW \ni R_G}{\mathrm{SO} \ni ID_{SO}, SO \ni (ID_{SO}||R_G)}$$

In order to achieve Goal 2, the SO recognize $\mathrm{M_2}$ and applies R1.

The SO sees some part of the message and compute

$$\frac{SO \ni Z_1, SO \ni ID_{SO}}{SO \ni V_1}$$

$$\frac{SO \ni S_1, SO \ni ID_{SO}, SO \ni S_1}{SO \ni V_2}$$

According to assumption A21, and GNY T1, P1, we obtain

$$\frac{SO \ni V_1, SO \ni V_2, SO \ni N_G}{, SO \ni K_S}$$

According to assumption A12, and GNY T1, P1, we obtain

$$\frac{\mathrm{GW} \triangleleft K_S}{\mathrm{GW} \ni K_S}$$

If the gateway node GW sees its own identity $\mathrm{ID_G}$, along with the message communicated, then GW believes $\mathrm{ID_G}$ and transmitted message.

$$\frac{\mathrm{GW} \triangleleft ID_G, SO \triangleleft \{(M_1\|M_2\|N_G\|R_G) \oplus Z_1\}_{R_{SO}}}{\mathrm{GW} \ni ID_G, SO \ni \{(M_1\|M_2\|N_G\|R_G) \oplus Z_1\}_{R_{SO}}}$$

The mobile GW possesses the identity of smart object $\mathrm{ID_{SO}}$ then we can represent it by the following way on holding message $(M_1\|M_2\|N_G\|R_G) \oplus Z_1$.

$$\frac{\mathrm{GW} \ni ID_{SO}}{\mathrm{GW} \ni ID_{SO}, SO \ni \{M_1\|M_2\|N_G\|R_G) \oplus Z_1\}_{R_{SO}}}$$

In order to achieve Goal 3, the GW recognizes K_S and applies R1

$$\frac{\text{GW}|\equiv\phi(V_1),\ \text{GW}|\equiv\phi(V_2),\ \text{GW}|\equiv\phi\ (N_G)\ ,\ \text{GW}\ni\phi(K_S)}{\text{GW}|\equiv\phi N_G}$$

Goal3 Achieved

According to assumption A9, and GNY T1, P1, we obtain

$$\frac{GW\ni V_1, GW\ni V_2, GW\ni N_G}{GW\ni K_S}$$

According to assumption A3, and GNY T1, P1, we obtain

$$\frac{\text{M}\lhd K_S}{\text{M}\ni K_S}$$

Finally, if the mobile M sees the identity of smart object ID_{SO}, along with the message communicated, then mobile believes ID_{SO} and transmitted message.

$$\frac{\text{M}\lhd ID_{SO}, M\lhd\{(M_1\|M_2\|N_G\|R_G\|T_2)\oplus ID_M\}_{R_M}}{\text{M}\ni ID_{SO}, M\ni\{(M_1\|M_2\|N_G\|R_G\|T_2)\oplus ID_M\}_{R_M}}$$

The mobile M possesses the identity of smart object ID_{SO} then we can represent it by the following way on holding message $(M_1\|M_2\|N_G\|R_G\|T_2)\oplus ID_M$.

$$\frac{\text{M}\ni ID_{SO}}{\text{M}\ni ID_{SO}, M\ni\{(M_1\|M_2\|N_G\|R_G\|T_2)\oplus ID_M\}_{R_M}}$$

In order to achieve Goal 4, the M recognizes K_S and applies R1

$$\frac{\text{M}|\equiv\phi(V_1),\ \text{M}|\equiv\phi(V_2),\ \text{M}|\equiv\phi\ (N_G)\ ,\ \text{M}\ni\phi(K_S)}{\text{M}|\equiv\phi N_G}$$

Goal4 Achieved

Therefore, all the three entities securely transmit the different credentials with each other and its honesty is confirmed by applying GNY logic.

B. PROVERIF2.03 SIMULATION

The issues of confidentiality, reachability, integrity, and the secrecy of all the credentials (secret keys, identity, random numbers, parameters, and timestamp) have been tested by using well-known software verification toolkit ProVerif2.03 [20]. The code and result are shown in appendix of the article.

C. INFORMAL SECURITY ANALYSIS

The security analysis of LMAS-SHS will informally be demonstrated as under:

1) RESISTS INSIDER ATTACK

As we do not prefer a storage table inside the gateway for secret credentials storage, an attacker cannot access the internal secrets. Similarly, the identity is secretly transmitted over the public network channel if, for example, an adversary copies a message from the open line due to random numbers different for each session, collision-free hash function, XOR operations, and nonce; they cannot theft any credentials to reach internally and hijack the system. Therefore, LMAS-SHS is free of insider threats.

2) WITHSTANDS TRACEABILITY ATTACK

The mobile, smart object and gateway node have first extracted nonce randomly and concatenated it with other credentials to make it secure. Due to this, an attacker cannot trace different sessions of the same system at different times. Therefore, LMAS-SHS is safe against traceability attacks.

3) RESISTS DOS ATTACK

Confirmation steps have been introduced in each round trip of the proposed protocol, i.e., confirms $\text{h}(\text{Z}_1||\text{ID}_\text{M})$, $\text{h}(\text{Z}_1||\text{ID}_\text{G})$, and $\text{h}(\text{Z}_1|\text{ID}_{\text{SO}})$. The checks can, in turn, mitigate denial of device attacks on the system. Similarly, after receiving the message by any participant, it first verifies the nonce received in, if successful, onward processing start, else, considered DoS attack from a potential attacker. Therefore, LMAS-SHS resists the DoS attack.

4) RESISTS REPLAY ATTACK

After receiving a message $\{\text{J}_1, \text{J}_2, \text{T}_1\}$ by a gateway, it first checks the timestamp (T_1) with the current timestamp (T_c) $\text{T}_1\text{-T}_\text{c}\leq\Delta\text{T}$, and if it is out of the pre-defined time threshold, the gateway considers it a potential replay attack, discard the message and does not proceed for further computation. Furthermore, if an attacker diverts $\{\text{M}_1, \text{M}_2, \text{T}_5, \text{M}_4, \text{M}_5, \text{N}_\text{G}\}$ message from the open network channel, the smart object also checks the timestamp (T_5) with its current time (Tc) to withstand with replay attack and vice versa, therefore, LMAS-SHS is safe against replay attacks.

5) WITHSTANDS MAN-IN-THE-MIDDLE ATTACK

Due to randomness in each transmitted message, the nonce is different for different sessions. Also, in the random extraction of large prime numbers, the adversary, if, for example, injects something new into the public network channel, they cannot do so due to no knowledge of N_M, N_G, and N_{SO}. Therefore, LMAS-SHS is robust against a man-in-the-middle attack.

$$\frac{\text{GW}|\equiv\phi(J_1),\ \text{GW}\ |\equiv\phi(N_M),\ \text{GW}\ |\equiv\phi\ (t)\ ,\ \text{GW}\ni\phi(S_5),\ \text{GW}\ni\phi(ID_M)}{\text{GW}|\equiv\phi N_M}$$

Goal1 Achieved

$$\frac{\text{SO}|\equiv\phi(M_1),\ \text{SO}\ |\equiv\phi(M_2),\ \text{SO}\ |\equiv\phi\ (N_G)\ ,\ \text{SO}\ni\phi(ID_G),\ \text{SO}\ni\phi(T),\ \text{SO}\ni\phi(Z_1)}{\text{SO}|\equiv\phi N_G}$$

Goal2 Achieved

6) FREE FROM DE-SYNCHRONIZATION ATTACK

There is no need to update parameters on the SO or M sides in LMAS-SHS. In contrast, in the case of some changes, each participant validates it correspondingly. Therefore, the SO, GW and M do not require synchronization properties in LMAS-SHS.

7) SUPPORT ANONYMITY

In LMAS-SHS, the GW uses anonymous identities for both SO, M, and itself, which means that the identity is untraceable. Also, two sessions are not stated with the same credentials due to random selection of the nonce (NG, NM, NSO) and timestamps. Therefore, LMAS-SHS supports the anonymity feature.

8) RESISTS STOLEN VERIFIER ATTACK

LMAS-SHS does not store any random number. The verification and validation of every credential do not require any database or tables on the M/SO side. Thus, if an adversary tries to reach internally to access the necessary certificates, they cannot masquerade as M or SO to mislead the GW in the authentication process. Therefore, LMAS-SHS resists stolen verifier attacks.

9) FREE FROM MASQUERADE ATTACKS

If an attacker uses a fake identity of any participant (ID_M, ID_{SO}, or ID_W) and tries to gain authorized access of the public channel, due to nonce, random numbers, and timestamp, any illegal attempt of an adversary will be denied by the system because of multiple checks in different round trip of the protocol. Therefore, LMAS-SHS is free from masquerade attacks.

IV. PERFORMANCE ANALYSIS

This paper section can be examined by considering computation, communication, and comparison analysis. These different performance metrics are as under:

A. COMPUTATION COSTS

In this section, we calculate the LMAS-SHS computation cost and compare it with state of the art scheme. We refer to the work of [16], [22] and [23] for a detailed comparison. The execution time of ECC point multiplication T_M is ($\sim$7.3529), hash function T_h is ($\sim$0.0004), fuzzy extractor T_R is ($\sim$7.3529), and encryption/decryption T_s is ($\sim$0.1303). The calculation of LMAS-SHS computation cost and detailed comparison with other protocols are shown in Table 2 and graphically in Fig. 2.

B. COMMUNICATION COST

We calculated the LMAS-SHS communication cost in this section and compared it with other schemes. We consider the work done in [16], [22] and [23] that defined encryption/decryption, ECC point; random number, hash function, timestamp and identity are {256, 320, 160, 160, 32, and 128}.

TABLE 2. Computation cost analysis.

Schemes	Total cost	Cost in ms
[16]	$12T_h$+$14T_M$+TR	90.2983
[17]	$15T_h$	4.0168ms
[18]	$16T_h$+$3T_m$	22.0651ms
[19]	$25T_h$+$4T_s$+$1T_R$	7.8841ms
[20]	$21T_h$+$8T_s$+$1T_R$	2.281ms
LMAS-SHS	$19T_h$+$4T_s$	0.5288ms

TABLE 3. Communication cost analysis.

Schemes	Number of Messages Communicated	Communication Costs in bits
[16]	3	2528
[17]	3	1484
[18]	4	1920
[19]	4	3360
[20]	4	2592
LMAS-SHS	4	1424

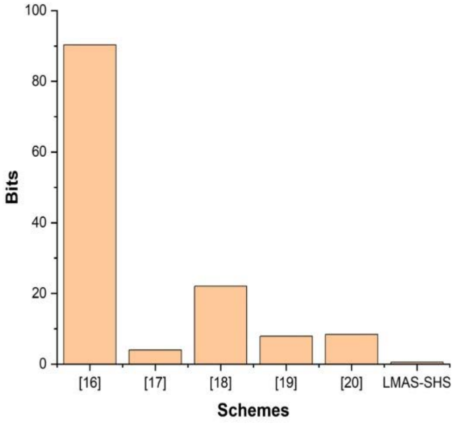


FIGURE 2. Computation cost in milliseconds (ms).

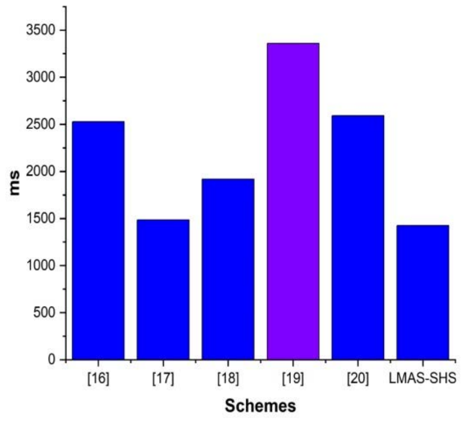


FIGURE 3. Communication cost in bits.

LMAS-SHS login and key exchange phase transmitted messages are Message1 = {J_1, J_2, T_1}, $Message_2$ = K_{EG}(M_1, T_3, M_2, M_3, M_4, M_5), $Message_3$ = {M_1, M_2, M_4, M_5, N_G},

TABLE 4. Security and functionalities comparison.

Security features↓	Schemes→	[16]	[17]	[18]	[19]	[20]	LMAS-SHS
Impersonation attack		✔	✔	✔	✔	✔	✔
Replay attack		✔	✔	✔	✔	✔	✔
Session key disclosure attack		✔	✔	✔	✔	✕	✔
MITM attack		✔	✔	✔	✔	✔	✔
Password guessing attack		✔	✔	✕	✔	✔	✔
Insider attack		✔	✔	✔	✕	✔	✔
Stolen device attack		✔	✔	✕	✕	✔	✔
Session temporary attack		✔	✔	✕	✕	✔	✔
Desynchronization attack		✔	✕	✔	✕	✔	✔
Mutual authentication		✔	✔	✔	✔	✔	✔
untraceability		✕	✔	✔	✔	✔	✔
Anonymity		✔	✔	✔	✕	✔	✔
Perfect forward secrecy		✔	✔	✕	✕	✔	✔

and $\mathrm{Message}_5 = \{M_1, M_2, M_4, N_G\}$ as shown in Table 3 and Fig. 3.

C. FUNCTIONALITIES COMPARISON

Suppose we compare LMAS-SHS with different protocols like [16]–[20] in terms of varying security functionalities/attacks. In that case, the proposed scheme resists all known attacks and is better than these schemes, as shown in Table 4. Whereas ✔ means Secure ✕ means insecure.

V. CONCLUSION

In this paper, we have presented a lightweight mutual authentication scheme for smart home surveillance called LMAS-SHS. In LMAS-SHS, different smart objects are fixed in various places for real-time information exchange towards the nearest gateway node regarding the health of the infrastructure and monitoring of the home. The ECC technique is used to design LMAS-SHS, which is lightweight and provides robust security. The security of LMAS-SHS has formally been verified using GNY logic and ProVerif2.03 and informally using pragmatic explanation. The performance analysis of LMAS-SHS is measured by considering computation and communication metrics. Consequently, the researchers have proved that LMAS-SHS is robust, lightweight, free from insider, stolen verifier attacks, and has no design flaw. So this is a more efficient protocol and provides security for smart home surveillance and can also be utilized for infrastructure inspection of a big city. Also, if this protocol is implemented for disaster purposes, it can quickly communicate the health of the whole infrastructure with the centralized server. Also, it can install in drone technology equipped with different smart objects near workers and employees for effective monitoring them to increase their work output and as an effective tool in the real estate business.

We plan to use Elliptic Curve Digital Signature Algorithm (ECDSA) to design a security mechanism for proving the transitional authentication of users in the teleworking environment. The security analysis of the said ECDSA-based transitional authentication scheme shall be manipulated via AVISPA (Automatic Validation of Internet Security Protocol Authentication). It is to mention that its performance will distress due to exponentiation in the discrete logarithmic function without affecting security.

APPENDIX

To check whether the session shared key is confidentiality communicated and whether it is reachable to each peer in an authentic manner, we used a verification toolkit ProVerif2.03. The result shows that an attacker at any stage could not crack the secrecy, confidentiality, and reachability of the session key.

```
 (*=======CHANNELS=======*)
 free CHS: channel [private].
 free CHP: channel.
 (*=======CONSTANT AND VARIABLES=======*)
 free NM:bitstring.
 free NG:bitstring.
 free NSO:bitstring.
 free RG:bitstring.
 free IDM:bitstring.
 free IDG:bitstring.
 free IDSO:bitstring.
 free KM:bitstring.
 free KG:bitstring.
 free KSO:bitstring.
 (*=======FUNCTIONS=======*)
 fun h(bitstring):bitstring.
 fun con(bitstring,bitstring):bitstring.
 fun XOR(bitstring,bitstring):bitstring.
 (*=======EQUATIONS=======*)
 equation forall a:bitstring, b:bitstring;
 XOR(XOR(a, b), b) = a.
 (*=======EVENTS=======*)
 event start_M(bitstring).
 event end_M(bitstring).
 event start_GW(bitstring).
 event end_GW(bitstring).
 event start_SO(bitstring).
 event end_SO(bitstring).
 (*=======QUERIES=======*)
 free KS:bitstring [private].
 query attacker(KS).
 query IDM:bitstring; inj-event(end_M(IDM)) ==>
 inj-event(start_M(IDM)).
 query IDG:bitstring; inj-event(end_GW(IDG)) ==>
```

```
inj-event(start_GW(IDG)).
query IDSO:bitstring; inj-event(end_SO(IDSO)) ==>
inj-event(start_SO(IDSO)).
(*=======PEER No. 01:MOBILE=======*)
let M =
(*........REGISTRATION SIDE COMPUTATION......*)
let PIDM = con(IDM,NM) in
let IM = con(PIDM,KM)) in
in(CHS, (S1:bitstring, S2:bitstring, S4:bitstring,
S5:bitstring, S6:bitstring, RG:bitstring));
let S1dash = XOR(NG, NM) in
let S2dash = XOR(S1, RG) in
let S4dash = h(con(con(S2, IDM), NM)) in
if S4dash = S4 then
let S5dash = h(con(con(IDM, KM), NM)) in
let S6dash = h(con(con(IDG, KG), NG)) in
out(CHS,(IM, IDM));
(*......AUTHENTICATION SIDE COMPUTATION........ *)
event start_M(IDM);
new T1:bitstring;
new Tc:bitstring;
let J1 = XOR(h(con(IDM, NM)), S4)) in
let J2 = XOR(h(con(con(con(J1, NM), T1), S5)), IDM)) in
out(C2,(J1, J2,T1));
in (CHP, (M1:bitstring, M2:bitstring, M4:bitstring,
T7:bitstring, RG:bitstring));
let Adash = h(con(S1, IDG)) in
if A = Adash then
let V1 = h(con(S1, IDM)) in
let V2 = h(con(S1, IDG)) in
let KS = h(con(con(V1, V2), NG)) in
event end_M(IDM)
else
0.
(*=======PEER No. 02: GATEWAY=======*)
let GW =
(*........REGISTRATION SIDE COMPUTATION......*)
in (CHS, (IM:bitstring, IDM:bitstring));
let S1 = XOR(NG, NM) in
let S2 = XOR(S1, RG) in
let S3 = h(con(S1, IDM)) in
let S4 = h(con(con(S1, IDM), NM)) in
let S5 = h(con(con(IDM, KM), NM)) in
let S6 = h(con(con(IDG, KG), NG)) in
out(CHS, (S1, S2, S4, S5, S6, RG));
in (CHS, (ISO:bitstring, IDSO:bitstring));
let Z1 = XOR(NG, NSO) in
let Z2 = XOR(Z1, RG) in
let Z3 = h(con(Z1, IDSO)) in
let Z4 = h(con(con(Z1, IDSO), NSO)) in
let Z5 = h(con(con(IDSO, KSO), NSO)) in
let Z6 = h(con(con(IDG, KG), NG)) in
out(CHS,(Z1, Z2, Z4, Z5, Z6, RG));
(*......AUTHENTICATION SIDE COMPUTATION........ *)
event start_WG(IDG);
in (CHP, (J1:bitstring, J2:bitstring, T1:bitstring));
new T2:bitstring;
new T3:bitstring;
new T6:bitstring;
new T7:bitstring;
new Tc:bitstring;
let M1 = XOR(XOR(S3, Z3), NG)) in
let M2 = XOR(XOR(IDG, NG), S1)) in
let M3 = XOR(XOR(IDM, NG), Z1)) in
let M4 = XOR(h(con(con(con(con(M1, M2), NG), RG), T2)),
S1) in
let M5 = XOR(h(con(con(con(con(M1. M2), NG), IDG),
T2)), Z1) in
out(CHP, (M1, M2, M3, M4, M5, T3));
in(CHP, (M1:bitstring, M2:bitstring, M4:bitstring,
M5:bitstring, T5:bitstring, RG:bitstring,));
let Adash = h(con(S1, IDG)) in
if Adash = A then
let V1 = h(con(con(S1, IDG), T6)) in
let V2 = h(con(Z1, IDG)) in
let KS = h(con(con(V1, V2), NG)) in
out(CHP, (M1, M2, M4, T7));
event end_U(IDG)
else
0.
(*=======PEER No. 03: SMART OBJECT=======*)
let SO =
(*.........REGISTRATION SIDE COMPUTATION.........*)
in(CHS, (S1:bitstring, S2:bitstring, S4:bitstring,
S5:bitstring, S6:bitstring, RG:bitstring));
let PIDSO = h(conIDSO, NSO)) in
let ISO = h(con(PIDSO, KSO)) in
out(CHS,(ISO, IDSO));
let Z1dash = XOR(NG, NSO) in
let Z2dash = XOR(Z1, RG) in
let Z4dash = h(con(con(Z2, IDSO), NSO)
if Z4dash = Z4 then
let Z5dash = h(con(con(IDSO, KSO), NSO)) in
let Z6dash = h(con(con(IDG, KG), NG)) in
(*.........AUTHENTICATION SIDE COMPUTATION.........*)
event start_SO(IDSO);
in(CHP, (M1:bitstring, M2:bitstring, M3:bitstring,
M4:bitstring, M5:bitstring, T3:bitstring));
let let Adash = h(con(S1, IDG)) in
if Adash = A then
new T4:bitstring;
let V1 = h(con(con(Z1, IDSO), T4)) in
let V2 = h(con(S1, IDSO)) in
let KS = h(con(con(V1, V2), NG)) in
out(CHP, (M1, M2, M4, M5, RG, T5));
event end_SO(IDSO)
else
0.
process ((!pSO) | (!pWG) | (!pM) )
The result indicated that the session key is much more
secure against any attack upon running the code. Its
confidentially and reachability are preserved as
shown below:
(*=======RESULT GENERATED=======*)
Completing equations...
Completing equations...
- Process 1- Query not attacker(KS[]) in process 1
Translating the process into Horn clauses...
Completing...
Starting query not attacker(SK[])
RESULT not attacker(KS[]) is true.
- Query inj-event(end_M(IDM[])) ==>
inj-event(start_M(IDM[])) in process 1
Translating the process into Horn clauses...
Completing...
Starting query inj-event(end_M(IDM[])) ==>
inj-event(start_M(IDM[]))
RESULT inj-event(end_M(IDM[])) ==>
inj-event(start_M(IDM[])) is true.
- Query inj-event(end_SO(IDSO[])) ==>
inj-event(start_SO(IDSO[])) in process 1
Translating the process into Horn clauses...
Completing...
Starting query inj-event(end_SO(IDSO[])) ==>
inj-event(start_SO(IDSO[]))
RESULT inj-event(end_SO(IDSO[])) ==>
inj-event(start_SO(IDSO[])) is true.
- Query inj-event(end_GW(IDG[])) ==>
inj-event(start_GW(IDG[])) in process 1
Translating the process into Horn clauses...
Completing...
Starting query inj-event(end_GW(IDG[])) ==>
inj-event(start_GW(IDG[]))
------------------------------------------------------
Verification summary:
Query not attacker(KS[]) is true.
Query inj-event(end_M(IDM[])) ==>
inj-event(start_M(IDM[])) is true.
Query inj-event(end_SO(IDSO[])) ==>
inj-event(start_SO(IDSO[])) is true.
Query inj-event(end_GW(IDG[])) ==>
inj-event(start_GW(IDG[])) is true.

------------------------------------------------------
```

REFERENCES

[1] S. U. Jan, S. Ali, I. A. Abbasi, M. A. A. Mosleh, A. Alsanad, and H. Khattak, "Secure patient authentication framework in the healthcare system using wireless medical sensor networks," *J. Healthcare Eng.*, vol. 2021, pp. 1–20, Jul. 2021.

[2] *Sumsung Smartthings Developers Documentation*. Accessed: Jan. 17, 2019. [Online]. Available: https://smartthings.developer.samsung.com/blog/en-us/2019/01/17/Shape-the-Future-of-IoTwith-SmartThings

[3] F. K. Santoso and N. C. H. Vun, "Securing IoT for smart home system," in *Proc. Int. Symp. Consum. Electron. (ISCE)*, Jun. 2015, pp. 1–2.

[4] S. Naoui, M. E. Elhdhili, and L. A. Saidane, "Lightweight and secure password based smart home authentication protocol: LSP-SHAP," *J. Netw. Syst. Manage.*, vol. 27, no. 4, pp. 1020–1042, Oct. 2019.

[5] Y. Zhang, D. He, L. Li, and B. Chen, "A lightweight authentication and key agreement scheme for Internet of Drones," *Comput. Commun.*, vol. 154, pp. 455–464, Oct. 2020.

[6] S. U. Jan, F. Qayum, and H. U. Khan, "Design and analysis of lightweight authentication protocol for securing IoD," *IEEE Access*, vol. 9, pp. 69287–69306, 2021.

[7] S. U. Jan, I. A. Abbasi, and F. Algarni, "A key agreement scheme for IoD deployment civilian drone," *IEEE Access*, vol. 9, pp. 149311–149321, 2021.

[8] J. Won, S.-H. Seo, and E. Bertino, "Certificateless cryptographic protocols for efficient drone-based smart city applications," *IEEE Access*, vol. 5, pp. 3721–3749, 2017.

[9] M. Wazid, A. K. Das, N. Kumar, A. V. Vasilakos, and J. J. P. C. Rodrigues, "Design and analysis of secure lightweight remote user authentication and key agreement scheme in Internet of Drones deployment," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3572–3584, Apr. 2018.

[10] W. Hong, L. Jianhua, L. Chengzhe, and W. Zhe, "A provably secure aggregate authentication scheme for unmanned aerial vehicle cluster networks," *Peer-to-Peer Netw. Appl.*, vol. 13, no. 1, pp. 53–63, Jan. 2020.

[11] S. U. Jan and H. U. Khan, "Identity and aggregate signature-based authentication protocol for IoD deployment military drone," *IEEE Access*, vol. 9, pp. 130247–130263, 2021.

[12] X. Li, T. Liu, M. S. Obaidat, F. Wu, and P. Vijayakumar, "A lightweight privacy-preserving authentication protocol for VANETs," *IEEE Syst. J.*, vol. 14, no. 3, pp. 3547–3557, May 2020.

[13] M. Gupta, M. Abdelsalam, S. Khorsandroo, and S. Mittal, "Security and privacy in smart farming: Challenges and opportunities," *IEEE Access*, vol. 8, pp. 34564–34584, 2020.

[14] R. Ali, A. Pal, S. Kumari, M. Karuppiah, and M. Conti, "A secure user authentication and key-agreement scheme using wireless sensor networks for agriculture monitoring," *Future Gener. Comput. Syst.*, vol. 84, pp. 200–215, Jul. 2018.

[15] L. Gong, R. Needham, and R. Yahalom, "Reasoning about belief in cryptographic protocols," in *Proc. IEEE Comput. Soc. Symp. Res. Secur. Privacy*, May 1990, pp. 234–248.

[16] S. Challa, M. Wazid, A. K. Das, N. Kumar, A. G. Reddy, E.-J. Yoon, and K.-Y. Yoo, "Secure signature-based authenticated key establishment scheme for future IoT applications," *IEEE Access*, vol. 5, pp. 3028–3043, 2017.

[17] J. Oh, S. Yu, J. Lee, S. Son, M. Kim, and Y. Park, "A secure and lightweight authentication protocol for IoT-based smart Homes," *Sensors*, vol. 21, no. 4, p. 1488, Feb. 2021.

[18] M. Shuai, N. Yu, H. Wang, and L. Xiong, "Anonymous authentication scheme for smart home environment with provable security," *Comput. Secur.*, vol. 86, pp. 132–146, Sep. 2019.

[19] M. Wazid, A. K. Das, V. Odelu, N. Kumar, and W. Susilo, "Secure remote user authenticated key establishment protocol for smart home environment," *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 2, pp. 391–406, Dec. 2017.

[20] N. Khan, J. Zhang, and S. U. Jan, "A robust and privacy-preserving anonymous user authentication scheme for public cloud server," *Secur. Commun. Netw.*, vol. 2022, pp. 1–14, Mar. 2022.

[21] B. Blanchet, B. Smyth, V. Cheval, and M. Sylvestre, "ProVerif 2.00: Automatic cryptographic protocol verifier, user manual and tutorial," Max-Planck Institut für Informatik, Saarbrücken, CNRS, Inria Paris, Paris, France, Tech. Rep. EPSRC project UbiVal (EP/D076625/2, 2018, pp. 5–16.

[22] A. Yazdinejad, G. Srivastava, R. M. Parizi, A. Dehghantanha, K.-K.-R. Choo, and M. Aledhari, "Decentralized authentication of distributed patients in hospital networks using blockchain," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 8, pp. 2146–2156, Aug. 2020.

[23] A. Yazdinejad, R. M. Parizi, G. Srivastava, A. Dehghantanha, and K.-K.-R. Choo, "Energy efficient decentralized authentication in internet of underwater things using blockchain," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2019, pp. 1–6.

SAEED ULLAH JAN received the Ph.D. degree in network security from the University of Malakand, in 2021. He is currently working as a Lecturer in computer science at the Higher Education, Achieves & Libraries Department, Government of Khyber Pakhtunkhwa, Pakistan. He is also working as a Controller of Examinations for nine B.S. Disciplines with the Government College Wari (Dir Upper), a far-flung remote area of the province where most of the youngsters have no access to universities/institutions for higher education. Furthermore, he has conducted research in many areas, including green computing, distributed computing, privacy-preserving parallel computation, and drone security and authentication. He has published over 25 research papers in prestigious conferences and journals and written an introductory book in computer science for beginners. The Government of Khyber Pakhtunkhwa, awarded "Best Teacher Award" for the year 2019–2020 out of 11000 College Teachers in 309 public sector colleges in the province.

IRSHAD AHMED ABBASI (Member, IEEE) received the M.S. degree in computer science from COMSATS University Islamabad, Pakistan, and the Ph.D. degree in computer science from Universiti Malaysia Sarawak, Malaysia. He worked as a Senior Lecturer at King Khalid University, Saudi Arabia, from 2011 to 2015. He is currently working as an Assistant Professor with the Computer Science Department, University of Bisha, Saudi Arabia. He was declared as the Best Teacher at the Faculty of Science and Arts Belqarn, University of Bisha, in 2016. He has over 12 years of research and teaching experience. He is the author of many articles published in top quality journals. His research interests include VANETs, MANETs, FANETs, mobile computing, the IoT, cloud computing, cybersecurity, soft computing, and drone security and authentication. He has received multiple awards, scholarships, and research grants. He is serving as an editor. He is also acting as a reviewer for many well reputed peer-reviewed international journals and conferences.

MOHAMMED A. ALQARNI received the bachelor's degree in computers from King Khalid University, Saudi Arabia, in 2008, the M.Sc. degree in computational sciences from Laurentian University, Sudbury, ON, Canada, in 2012, and the Ph.D. degree in computer science from McMaster University, Hamilton, ON, Canada, in 2016. He is currently an Associate Professor at the College of Computer Science and Engineering, University of Jeddah, Saudi Arabia.