IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 15, 2021, accepted December 8, 2021, date of publication December 21, 2021,
date of current version January 4, 2022.

*Digital Object Identifier 10.1109/ACCESS.2021.3137175*

# Lightweight and Privacy-Preserving Remote User Authentication for Smart Homes

**K. NIMMY[1], (Student Member, IEEE), SRIRAM SANKARAN[1], (Member, IEEE), KRISHNASHREE ACHUTHAN[1], (Senior Member, IEEE), AND PRASAD CALYAM[2], (Senior Member, IEEE)**

[1]Center for Cybersecurity Systems and Networks, Amrita Vishwa Vidyapeetham, Amritapuri 690525, India
[2]Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, MO 65211, USA

Corresponding author: K. Nimmy (nimmy@am.amrita.edu)

The work of K. Nimmy was supported by the Ministry of Electronics and Information Technology (MeitY), Government of India, under the Visvesvaraya Ph.D. Scheme for Electronics and IT.

**ABSTRACT** The rapid proliferation of embedded devices has led to the growth of the Internet of Things (IoT) with applications in numerous domains such as home automation, healthcare, education and agriculture. However, many of the connected devices particularly in smart homes are the target of attacks that try to exploit security vulnerabilities such as hard-coded passwords and insecure data transfer. Recent studies show that there is a considerable surge in the number of phishing attacks targeting smart homes during the COVID-19 pandemic. Moreover, many of the existing user authentication protocols in the literature incur additional computational overhead and need to be made more resilient to smart home targeted attacks. In this paper, we propose a novel lightweight and privacy-preserving remote user authentication protocol for securing smart home applications. Our approach is based on Photo Response Non-Uniformity (PRNU) to make our protocol resilient to smart home attacks such as smartphone capture attacks and phishing attacks. In addition, the lightweight nature of our solution is suitable for deployment on heterogeneous and resource constrained IoT devices. Besides, we leverage geometric secret sharing for establishing mutual authentication among the participating entities. We validate the security of the proposed protocol using the AVISPA formal verification tool and prototype it on a Raspberry Pi to analyze the power consumption. Finally, a comparison with existing schemes reveals that our scheme incurs a 20% reduction in communication overhead on smart devices. Furthermore, our proposed scheme is usable as it absolves users from memorizing passwords and carrying smart cards.

**INDEX TERMS** Authentication protocol, Internet of Things, protocol resiliency, geometric secret sharing, smart home, PRNU.

## I. INTRODUCTION

The ubiquity and increasing popularity of Internet of Things (IoT) has led to the proliferation of embedded devices. The number of these devices was predicted to be around 400 million at the end of 2016 and it is projected to reach 1.5 billion in 2022 [1]. The potential application domains of IoT include environmental monitoring, energy and water management, smart cities, healthcare and supply chain management. Moreover, the IoT paradigm has the potential to address economic and environmental needs particularly in smart homes. Thereby, IoT integration in smart homes massively contributes to several sustainable development goals, through its capacity to increase efficiency and save costs.

However, the security aspects of the IoT devices in smart homes are not always covered holistically thus, making them vulnerable to cyber-attacks. Recently, researchers discovered security-critical design flaws in smart home devices [2]; in particular, devices using ZigBee and ZWave- wireless communication protocols for the smart homes [3], [4]. Numerous vulnerabilities have been identified in the OAuth protocol [5], which is the de-facto protocol used for authentication and authorization in the smart homes. Moreover, studies show

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

that the remote work and remote schooling due to COVID-19 have a multiplier effect on the rise of IoT attacks in smart homes. The number of phishing attacks aimed to steal the user credentials are on the rise in the past one year [6]. Besides, the heterogeneous and non-standardized architecture of IoT result in a greater number of smart home attacks [7].

Vulnerable smart home devices are increasingly being targeted by attackers to steal personal information and also launch distributed denial-of-service (DDoS) attacks [8], [9]. For instance, in October 2016, a massive Mirai botnet attack was launched, which almost brought down the Internet by taking advantage of vulnerable smart devices [10]. The Mirai botnet code uses telnet service to find devices such as smart home routers, security cameras, DVRs, etc., that are still using their factory default username and password. Nevertheless, numerous variants of Mirai have emerged in the last two years, which can infect many types of devices [11]. A security breach in smart homes can have high impact because it allows attackers to take control of the devices in smart homes, steal sensitive information and blackmail the occupants at very large scales [12].

In this paper, we propose a novel lightweight and privacy-preserving remote user authentication protocol for smart home environments. Our proposed protocol is based on geometric secret sharing and uses face biometric and Photo Response Non-Uniformity (PRNU) [13] to authenticate both users and their smartphones. Unlike multi-factor authentication schemes, our scheme requires users to provide a single factor to prove the identity of both user and smartphones. Face biometric alone may not be sufficient to provide security as it is susceptible to face spoofing attacks [14]. Moreover, existing studies show that PRNU of a smartphone camera can be used to uniquely identify the device with an error rate less than 0.5% [15]. Hence, we incorporate PRNU-based smartphone authentication of the users to increase the complexity of the attackers. We conduct experiments to prove the effectiveness of using PRNU for smartphone authentication using 100 face images collected from 10 different smartphones. Further, the conventional peak-to-correlation energy (PCE) of the PRNU of the authenticated smartphone and the PRNU obtained from another set of face images is computed. Besides, we leverage geometric secret sharing to establish mutual authentication among the user, gateway and IoT device. A high-level representation of the mutual authentication between the user and the gateway is depicted in Fig. 1. Geometric secret sharing allows two entities to share completely distinct shares of the secret which are then combined to retrieve the secret. Further, compromise of any of these shares neither reveals the other nor the secret as it is impossible to determine the line given a single point (share). Moreover, the secret reconstruction operation contains lightweight operations such as addition and subtraction. Hence, geometric secret sharing based mutual authentication provides better security than password or shared key based schemes to protect the smart devices.

Further, we prototype the proposed protocol in a Raspberry Pi and measure power and energy consumption. The proposed protocol uses simple hash and XOR operations at the device side to make it lightweight. Moreover, formal analysis and security properties verification have been done to prove that the proposed protocol is resilient to various known attacks.

### A. OUR CONTRIBUTIONS

Our contributions to this paper are listed as follows:

1) We propose a novel lightweight and privacy-preserving remote user authentication protocol for smart home environments based on geometric secret sharing. The proposed protocol builds on a smart home threat model and leverages PRNU to uniquely identify the smartphone of the user.
2) To our best of knowledge, our proposed lightweight protocol is the first to achieve mutual authentication using geometric secret sharing.
3) We conduct experiments to show the effectiveness of PRNU in uniquely identifying the smartphone of the users by collecting 100 face images from 10 different smartphones.
4) Security analysis using AVISPA tool and performance analysis show that the proposed protocol is highly secure against common attacks and there is a 20% of reduction in communication overhead at the smart device compared to the existing state-of-the-art schemes.

The remainder of the paper is organized as follows: Section 2 presents related work. Section 3 provides a background on smartphone camera identification and mutual authentication using geometric secret sharing. Section 4 details our proposed protocol along with the threat model. In Section 5, we present a formal security analysis of our protocol. Section 6 details performance evaluation experiments and results. Section 7 concludes the paper.

## II. RELATED WORK

A robust authentication method based on biometric identifiers can be considered as an effective countermeasure for tackling security risks related to IoT devices in smart homes [16]. Most of the existing protocols are designed based on passwords for user authentication [17], which make them susceptible to shoulder surfing attack in which an adversary observes directly over the shoulders or use external recording devices such as CCTV camera to collect users' credentials [18]. A passwordless authentication scheme is more convenient for users and simultaneously makes the tasks of the attackers difficult. Another essential factor in the design of an authentication scheme is its resilience to social engineering attacks [19], where psychological manipulations are used to trick users into making security mistakes or giving away sensitive information. An authentication scheme based on users' biometric identity can eliminate social engineering attacks to a significant extent as the credentials cannot be

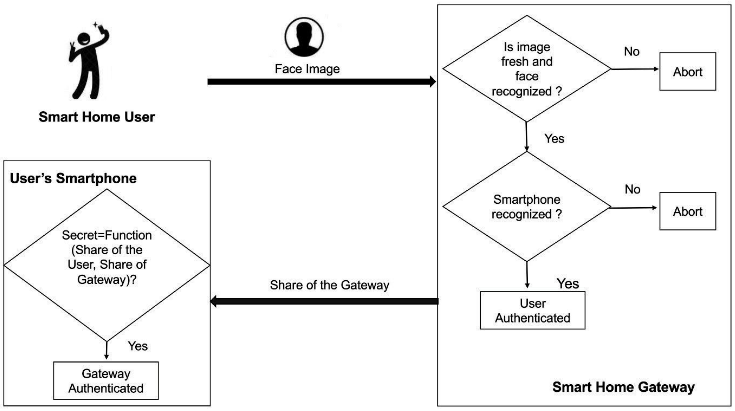


**FIGURE 1. Authentication among users, smartphones and the gateway nodes leveraging face biometric and PRNU for user and smartphone authentication respectively and geometric secret sharing for mutual authentication between user and the gateway.**

forged by an attacker using psychological tricks. Further, authentication based on the fingerprint of the smartphone is one of the effective approaches to authenticate a user to verify the possession of a registered smartphone [20].

Various authentication schemes in prior works require additional devices such as smart cards. Moreover, most of the existing user authentication approaches in the literature are susceptible to password guessing attacks, smartphone capture attacks, smart device capture attacks, user impersonation attacks and shoulder surfing attacks [15], [21]–[24]. Besides, conventional security mechanisms have become inapplicable as many of the IoT devices are resource-constrained and heterogeneous in terms of underlying communication protocols, data formats and technologies [25]. The above factors necessitate the development of lightweight, privacy-preserving and secure solutions for smart homes [26], which is the focus of our novel work in this paper.

While the working environment shifted to work from home (WFM) scenario due to the COVID-19 pandemic, cyber-attacks on individuals and organizations continue to rise steadily. The attackers are taking advantage of this scenario as the employees have to use their personal devices in the home network, lacking sufficient security measures [27]. Studies report that 76% of organizations are unprepared for facing security challenges [28]. Moreover, the attackers are launching phishing attacks disguised in the form of legitimate authorities with links pertaining to COVID-19 [29]. For instance, in Italy, cybercriminals sent emails aimed at infecting user's computer, disguising as World Health Organization [30]. Phishing emails may enable attackers to gain access to organizations' networks that was never intended for the public through unprotected devices in the home network. The pandemic witnessed various attacks ranging from online meeting hijacking, phishing, malware, ransomware and fake apps [31].

Recently, many user authentication schemes have been proposed for smart homes. Wazid *et al.* [21] proposed a shared key-based remote user authentication protocol for the smart home environment. The computation cost for their scheme is more than many of the schemes they considered for comparison. Besides, the scheme leaves out security features such as resilience to smart device capture attacks. A session key establishment scheme for the smart home environment was proposed by Kumar *et al.* [32]. This scheme does not support forward secrecy, gateway anonymity and mutual authentication. Further, Kumar *et al.* [33] developed a framework for connected smart home environments. The authors claim that their scheme provides anonymity and unlinkability, which make network tracking difficult in smart home networks. However, their scheme is vulnerable to physical attack in which an attacker tries to read out the keys stored in the smart devices. Oh *et al.* [34] proposed a password-based authentication protocol for smart homes. Their proposed scheme is vulnerable to password guessing attack and incurs high communication and computation cost than our proposed scheme. Similarly, the scheme proposed by Fakroon *et al.* [35] also incurs high communication cost than our proposed scheme. Public key based authentication protocols [22], [36]–[39] are computationally expensive than symmetric key based approaches. Besides, almost all of the existing schemes are designed using timestamps, making them vulnerable to clock synchronization problems [21], [32], [36], [40]. In particular, IoT devices are more prone to such problems as they are deployed in a broad range of operating conditions and are resource-constrained [41].

In contrast to the above-mentioned schemes, we propose a hybrid key-based (uses both public key cryptography and symmetric key cryptography) lightweight and privacy-preserving remote user authentication scheme that uses geometric secret sharing for establishing mutual authentication. The proposed protocol alleviates the burden of users from remembering long passwords and carrying additional mechanisms for authentication. PRNU and biometric based user authentication make the protocol resilient to spoofing and phishing attacks. Geometric secret sharing based mutual authentication among the participating entities prevents fake gateway and smart device impersonation attacks.

## III. BACKGROUND

### A. SMARTPHONE CAMERA IDENTIFICATION

The Photo-Response Non-Uniformity (PRNU) of an image is widely used in source camera identification (SCI) [42]. PRNU fingerprint of the source camera is mainly caused by imperfections that occurred during the sensor manufacturing process and different sensitivity of pixels to light [43]. The process of extracting PRNU fingerprint from an image for source camera identification is as follows. The digital camera output can be modeled as (1).

$$I_r = I^0 + I^0 K + \theta \tag{1}$$

where $I^0$ is the noise-free version of the image $I_r$, $K$ is the camera PRNU fingerprint and $\theta$ represents a combination of independent random noises. Firstly, a denoising filter $F$ such as weiner2 (in MATLAB) is applied to the image $I_r$ and it is subtracted from the image to obtain the noise residue as given in (2).

$$Q = I_r - F(I_r) \tag{2}$$

where $Q$ is the image noise residual. The PRNU fingerprint, $\hat{K}$, is derived from $N$ number of images by computing the maximum likelihood estimate as given in (3) :

$$\hat{K} = \frac{\sum_{i=1}^{N} Q_i I_{r_i}}{\sum_{i=1}^{N} (I_{r_i})^2} \tag{3}$$

The estimated fingerprint $\hat{K}$ contains a small noise factor $\delta$ and is represented as $\hat{K} = K + \delta$, where $K$ is the real fingerprint. To identify the source camera of an image, a correlation, $corr(K, \hat{K})$ of the estimated fingerprint and the real fingerprint is computed.

### B. MUTUAL AUTHENTICATION USING GEOMETRIC SECRET SHARING

In a geometric secret sharing scheme, a secret is split into shares to distribute among the participants in such a way that authorized subsets of participants alone can reconstruct the secret. A $(t, n)$ threshold secret sharing scheme proposed by Shamir [44] and Blakley and Kabatianskii [45] distributes shares amongst $n$ participants and allows only $t$ participants to recover the secret $S_{ec}$ but no group of fewer than $t$ participants can. The essential idea is that two points are sufficient to define a line in which the secret $S_{ec}$ is the coordinates of a fixed point $(0, S_{ec})$ on a given line $l$, which intersects the $y$ axis at the secret point [46]. Let $(1, S_{ec} + R_{and})$ and $(2, S_{ec} + 2R_{and})$ be the coordinates of line $l$ where $R_{and}$ is a random slope. Given any two points, the line $l$ can be determined and the $y$-intercept of this line represents the secret. This is a $(2, n)$ threshold scheme for any $n$. We leverage this scheme establish mutual authentication among participating entities. The shares can be created as given in (4).

**TABLE 1. Notations used in the proposed protocol.**

| Notations | Description |
|---|---|
| $U_i, G, D_j$ | $i^{th}$ user, gateway and $j^{th}$ device |
| $ID_i, ID_G, ID_j$ | Identities of user, gateway and device |
| $TD_x$ | Temporal identity of $x$ |
| $K_{U_i}, K_G, K_{D_j}$ | Public keys of user, gateway and device |
| $S_i, S_j$ | Secrets generated for user and device |
| $s_i, s_j$ | Shares generated by user and device |
| $s_{G_i}, s_{G_j}$ | Shares generated by $G$ for user and device |
| $K_{UiG}$ | Pre-shared keys with user and gateway |
| $R$ | Random number |
| $N_x$ | Random nonce generated by $x$ |
| $h(.)$ | Hash function |
| $K_S$ | Session key |
| $p$ | Prime number |
| $P_{rnu}$ | Photo response non-uniformity |
| $I$ | Face image |
| $I_m$ | Metadata of the face image |
| $\oplus$ | Bitwise XOR |
| $RREG$ | Re-registration |

$$\begin{aligned} share_1 &= (S_{ec} + R_{and}) \ mod \ p \\ share_2 &= (S_{ec} + 2R_{and}) \ mod \ p \end{aligned} \tag{4}$$

where $p$ is a prime number. The shares $share_1$ and $share_2$ are then combined to reconstruct the secret $S_{ec}$ as given in (5).

$$S_{ec} = (2 \times share_1 - share_2) \ mod \ p \tag{5}$$

Two points (shares) are sufficient to determine a line that passes through both the points. Once line is determined, we can find the intersection of the line with the $y$ axis which is the secret. Even though one of these shares (points) is compromised, finding the other share (point) is impossible as there are infinitely many lines passing through a point.

## IV. PROPOSED PROTOCOL

We proposed a preliminary version of a user authentication scheme based on public-key cryptography and conducted a performance evaluation [37]. As an extension to the proposed work, we developed a lightweight version of [37] and conducted a power analysis of the protocol. The enhanced version of the proposed protocol is hybrid key-based as it uses asymmetric and symmetric keys. Based on the power analysis results, we use asymmetric keys for generating temporal identities. Besides, the proposed scheme uses simple hash and XOR operations to establish authentication and key agreement. Hence, the proposed protocol in this paper has lower computation and communication overhead than the existing protocols. The novelty of the scheme is based on how efficiently and securely the proposed protocol achieves mutual authentication and key establishment among all participating entities using geometric secret sharing resulting in minimal computation and communication costs. Besides, the proposed scheme achieves user and smartphone authentication leveraging a single face image.

In this section, we discuss the proposed novel lightweight and privacy-preserving remote user authentication protocol for smart home environments using geometric secret sharing. The proposed protocol builds on a smart home threat model we developed and consists of four phases: initialization phase, registration phase, authentication and key establishment phase and re-registration phase. Table 1 shows the notations used in the proposed protocol.

## A. SMART HOME THREAT MODEL

### 1) HOME NETWORK MODEL

The home network model for the proposed protocol, is depicted in Fig. 2. The participating entities include users, smart devices and gateway. To start with, a user needs to authenticate to the gateway to access devices in the home network. Generally, a smart home network consists of $n$ number of users of the smart home denoted as $U_i$ where $i = 1..n$, $m$ number of smart devices $D_j$ where $j = 1..m$ and the gateway $G$. The smart devices are connected to the internet through the gateway. The gateway handles the registration and authentication of the users and smart devices. All requests pass through the gateway, verifying the authenticity before it is sent to the smart devices. To access a device $D_j$ in the network, the user $U_i$ sends a request to the gateway $G$. $G$ verifies the authenticity of the user $U_i$ and sends messages to both $U_i$ and the device $D_j$. These messages help to authenticate the gateway $G$ to the user $U_i$ and the device $D_j$. According to our proposed scheme, only an authenticated user $U_i$ and device $D_j$ can compute the session key and communicate with each other.

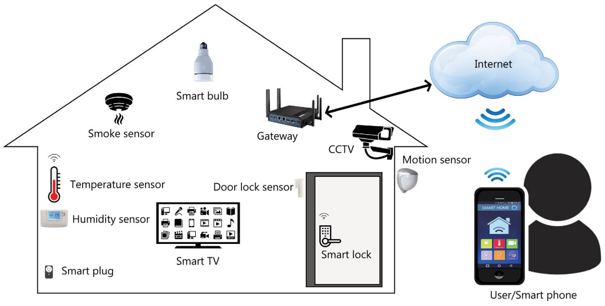


**FIGURE 2. Remote user authentication in smart home environment.**

### 2) THREAT MODEL

We use the formal Dolev-Yao threat model [47] for the proposed protocol for smart home environments. The communicating user or the smart device are not considered to be trusted. According to this model, the adversary who is an active eavesdropper is assumed to have access to the messages passing through the network. The adversary $A$ can:

- $A$ can actively eavesdrop on the channel to obtain messages passing through the channel.
- $A$ can send, delete and modify messages.
- $A$ can replay the messages to prevent the protocol from achieving its goals.

We also consider the following smart home specific attacks:

- $A$ can install a fake gateway.
- $A$ can fake smart device.
- $A$ can capture a smart device and perform side-channel attacks to extract sensitive information stored in that device.

## B. ASSUMPTIONS

We make use of the following assumptions for the proposed protocol.

- The devices are assumed to be registered at the gateway in off-line mode.
- The gateway is assumed to be completely trusted and protected from adversaries during the registration phase.
- The face recognition method is assumed to be resilient to three types of face spoofing attacks, namely printed photo attack, printed mask attack and displayed video attack on mobile phone/HD screen [48].
- Keys are assumed to be stored in the secure key storage area in the internal memory of the smartphones [49].
- Users' smartphones are secured using a local authentication mechanism such as fingerprint authentication.

## C. INITIALIZATION PHASE

In this phase, the public, private keys are generated using Elliptic Curve Cryptography (ECC). For this purpose, the gateway selects an Elliptic curve $E(F_p)$ over a finite field $F_p$ which is defined by the equation $y^2 = x^3 + ax + b \ mod \ p$ with a generator $P$ where $a, b \in F_p$. Users and devices choose a private key $n_A$ and then compute the public key as $K_A = n_A.P$. Symmetric keys (pre-shared keys) are also generated and shared between the user and the gateway. Besides, users register their face images at the gateway at this phase. The gateway uses this information for recognizing the users during the subsequent phases.

## D. USER REGISTRATION PHASE

In this phase, the user registers with the gateway providing a newly captured face image and choosing a unique identity. The smartphone encrypts the image using the pre-shared key and sends it to the gateway. Upon receiving the message, the gateway performs image analysis (*iminfo* in MATLAB) to check the freshness of the received image. After successful verification, the gateway performs face recognition and extracts PRNU $P_{rnu}$ from the face image and stores it along with the metadata $I_m$ of the image for further verification. Further, a secret is established using freshly generated nonce values exchanged between the user and the gateway. The secret is then split into two shares at both ends using a randomly chosen value. At the end of a successful run of the protocol, shares are created and stored at each end, later used for mutual authentication. Besides, the identity of the gateway will be provided by the mobile application. The detailed steps of the registration phase, as depicted in Fig. 3, are listed as follows.

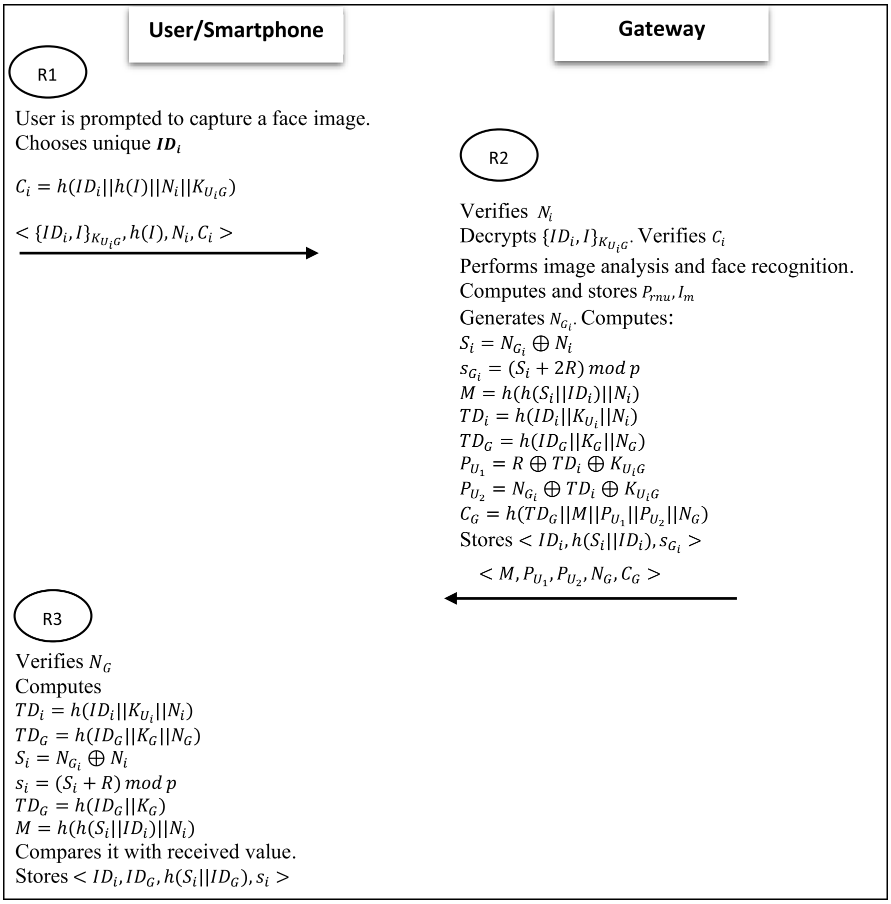

User/Smartphone
Gateway
R1
User is prompted to capture a face image.
Chooses unique $ID_i$
$C_i = h(ID_i||h(I)||N_i||K_{U_iG})$
$< \{ID_i, I\}_{K_{U_iG}}, h(I), N_i, C_i >$
R2
Verifies $N_i$
Decrypts $\{ID_i, I\}_{K_{U_iG}}$. Verifies $C_i$
Performs image analysis and face recognition.
Computes and stores $P_{rnu}, I_m$
Generates $N_{G_i}$. Computes:
$S_i = N_{G_i} \oplus N_i$
$s_{G_i} = (S_i + 2R)\ mod\ p$
$M = h(h(S_i||ID_i)||N_i)$
$TD_i = h(ID_i||K_{U_i}||N_i)$
$TD_G = h(ID_G||K_G||N_G)$
$P_{U_1} = R \oplus TD_i \oplus K_{U_iG}$
$P_{U_2} = N_{G_i} \oplus TD_i \oplus K_{U_iG}$
$C_G = h(TD_G||M||P_{U_1}||P_{U_2}||N_G)$
Stores $< ID_i, h(S_i||ID_i), s_{G_i} >$
$< M, P_{U_1}, P_{U_2}, N_G, C_G >$
R3
Verifies $N_G$
Computes
$TD_i = h(ID_i||K_{U_i}||N_i)$
$TD_G = h(ID_G||K_G||N_G)$
$S_i = N_{G_i} \oplus N_i$
$s_i = (S_i + R)\ mod\ p$
$TD_G = h(ID_G||K_G)$
$M = h(h(S_i||ID_i)||N_i)$
Compares it with received value.
Stores $< ID_i, ID_G, h(S_i||ID_G), s_i >$

**FIGURE 3.** User registration at the smart home gateway.

*Step R1:* User $U_i$ is prompted to capture a face image $I$ and chooses a unique identity $ID_i$ and enters it into the smartphone. The smartphone generates a nonce $N_i$ and computes $C_i = h(ID_i||h(I)||N_i||K_{U_iG})$. Smartphone encrypts $ID_i$ and $I$ using the pre-shared key $K_{U_iG}$ and sends the message $< \{ID_i, I\}_{K_{U_iG}}, h(I), N_i, C_i >$ to the gateway.

*Step R2:* Upon receiving the message, gateway verifies the nonce $N_i$. Upon successful verification, the gateway decrypts the message $\{ID_i, I\}_{K_{U_iG}}$ and computes $C_i = h(ID_i||h(I)||N_i||K_{U_iG})$ to verify the integrity of the message and verifies $h(I)$. Gateway then verifies the freshness of the image using image analysis and upon verification, performs face recognition. Upon successful verification, gateway extracts $P_{rnu}$ and metadata $I_m$ and stores it for future verification. In order to establish a secret $S_i$, the gateway generates a random nonce $N_G$ and a random number $R$. Gateway computes the secret as $S_i = N_{G_i} \oplus N_i$ and its share $s_{G_i} = (S_i + 2R) mod\ p$. To ensure the integrity of the secret computed, it computes $M = h(h(S_i||ID_i)||N_i)$. To send the random number $R$ and nonce $N_G$ to the user, gateway computes $TD_i = H(ID_i||K_{U_i}||N_i)$ and encrypts $R$ as $P_{U_1} = R \oplus TD_i \oplus K_{U_iG}$, similarly encrypts $N_G$ as $P_{U_2} = N_G \oplus TD_i \oplus K_{U_iG}$. Further, gateway generates another nonce $N_G$ and computes $TD_G = h(ID_G||K_G||N_G)$ and $C_G = h(TD_G||M||P_{U_1}||P_{U_2}||N_G)$ for verification at the user end. Gateway sends $< M, P_{U_1}, P_{U_2}, N_G, C_G >$ to the user and then stores $< ID_i, h(S_i||ID_i), s_{G_i} >$ in its memory for future verification.

*Step R3:* Upon successful verification of the nonce $N_G$, the smartphone computes the temporal identities $TD_i = H(ID_i||K_{U_i}||N_i)$ and $TD_G = h(ID_G||K_G||N_G)$. Smartphone then computes $C_G = h(TD_G||M||P_{U_1}||P_{U_2}||N_G)$ and verifies the received value. Upon successful verification, smartphone extracts $R = P_{U_1} \oplus TD_i \oplus K_{U_iG}$ and $N_G = P_{U_2} \oplus TD_i \oplus K_{U_iG}$ and then computes the secret $S_i = N_i \oplus N_G$. Smartphone then proceeds to compute its share $s_i = (S_i + R) mod\ p$ and stores it for future verification. To verify the correctness of the secret computed, the smartphone computes $M = h(h(S_i||ID_i)||N_i)$ and verifies it with the received value. At this point, the smartphone is assured that the secret $S_i$ is computed successfully at both ends. Smartphone stores $< ID_i, ID_G, h(S_i||ID_G), s_i >$ in its memory. Besides, the public keys of the gateway and the device will be shared by the mobile application after a successful registration.

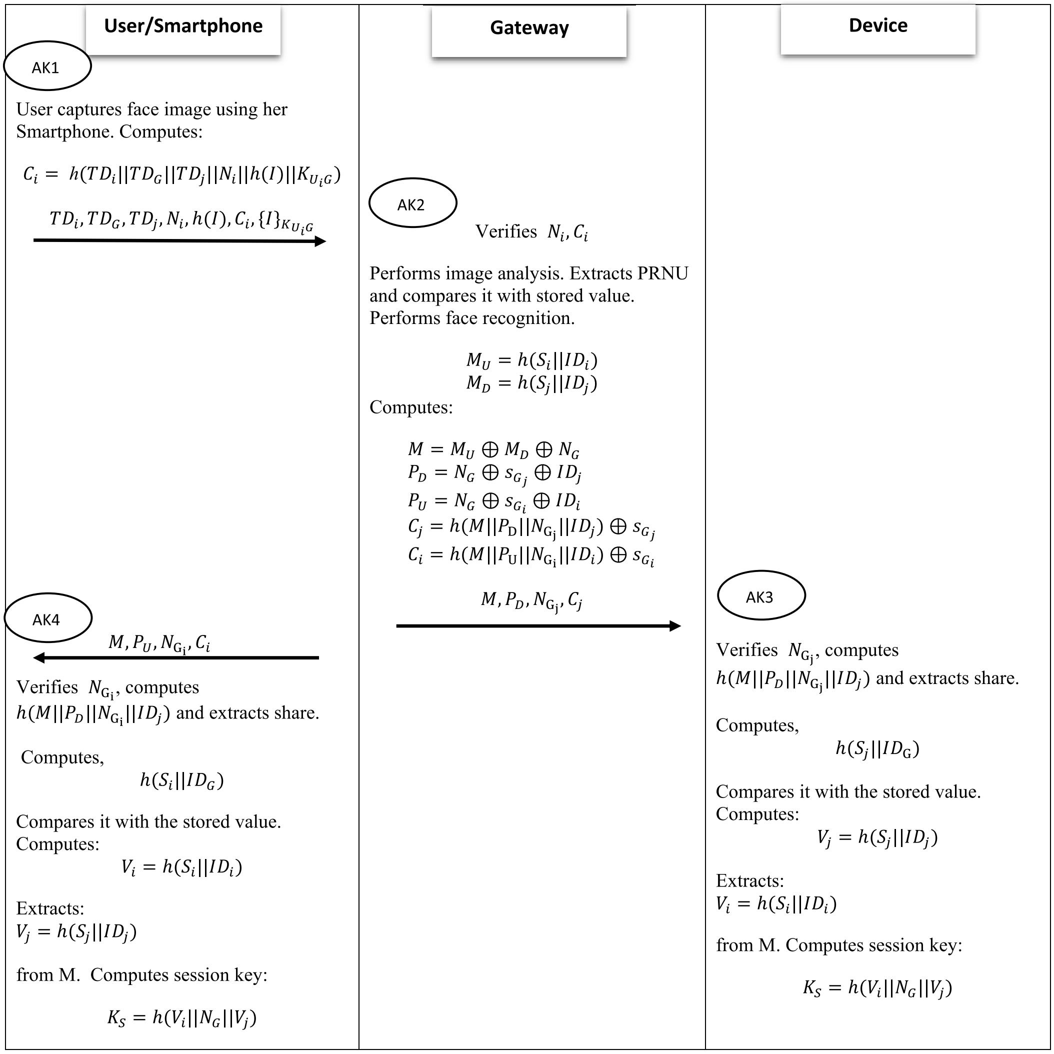

User/Smartphone
Gateway
Device
AK1
User captures face image using her Smartphone. Computes:
$C_i = h(TD_i||TD_G||TD_j||N_i||h(I)||K_{U_iG})$
$TD_i, TD_G, TD_j, N_i, h(I), C_i, \{I\}_{K_{U_iG}}$
AK2
Verifies $N_i, C_i$
Performs image analysis. Extracts PRNU and compares it with stored value. Performs face recognition.
$M_U = h(S_i||ID_i)$
$M_D = h(S_j||ID_j)$
Computes:
$M = M_U \oplus M_D \oplus N_G$
$P_D = N_G \oplus s_{G_j} \oplus ID_j$
$P_U = N_G \oplus s_{G_i} \oplus ID_i$
$C_j = h(M||P_D||N_{G_j}||ID_j) \oplus s_{G_j}$
$C_i = h(M||P_U||N_{G_i}||ID_i) \oplus s_{G_i}$
$M, P_D, N_{G_j}, C_j$
AK3
Verifies $N_{G_j}$, computes $h(M||P_D||N_{G_j}||ID_j)$ and extracts share.
Computes, $h(S_j||ID_G)$
Compares it with the stored value.
Computes: $V_j = h(S_j||ID_j)$
Extracts: $V_i = h(S_i||ID_i)$
from M. Computes session key:
$K_S = h(V_i||N_G||V_j)$
AK4
$M, P_U, N_{G_i}, C_i$
Verifies $N_{G_i}$, computes $h(M||P_D||N_{G_i}||ID_j)$ and extracts share.
Computes, $h(S_i||ID_G)$
Compares it with the stored value.
Computes: $V_i = h(S_i||ID_i)$
Extracts: $V_j = h(S_j||ID_j)$
from M. Computes session key:
$K_S = h(V_i||N_G||V_j)$

**FIGURE 4.** User and device authentication and key establishment using geometric secret sharing.

### E. AUTHENTICATION AND KEY ESTABLISHMENT PHASE

This phase, depicted in Fig. 4, is invoked when the user wants to access a device in the smart home. The user is then prompted to capture a face image and it is sent to the gateway for further processing. The gateway ensures the freshness of the image by performing image analysis. On successful verification, the photo response non-uniformity (PRNU) and image metadata are then extracted and compared with the stored values. Upon successful verification, the gateway performs face recognition and then it retrieves the hash of the secrets pertaining to the user and the device and sends it to the device and the user along with its share. The user and the device use this information for establishing a session key. The steps for this phase are described as follows.

*Step AK*1*:* User is prompted to capture her face image and selects the device, the smartphone computes the hash of the image and generates a nonce $N_i$. Further, smartphone computes $C_i = h(TD_i||TD_G||TD_j||N_i)$ where $TD_i = h(ID_i||K_{U_i}||N_i)$, $TD_G = h(ID_G||K_G||N_i)$ and $TD_j = h(ID_j||K_{D_j}||N_i)$. It then sends the message $< TD_i, TD_G, TD_j, h(I), N_i, C_i, \{I\}_{K_{U_iG}} >$ to the gateway where $TD_j$ is the temporal identity of the device the user wants to access.

*Step AK*2*:* After verification of $N_i$ and $C_i$, the gateway decrypts the image $I$ and verifies integrity and freshness of the image by performing hash comparison and image analysis respectively. Upon successful verification, it extracts $P_{rnu}$ and $I_m$ and compares with the stored values. Gateway then proceeds to perform face recognition of the user. After successful verification, the gateway retrieves $M_D = h(S_j||ID_j)$ and $M_U = h(S_i||ID_i)$ from its memory. To establish a session key at both user and device end, Gateway computes $M = M_U \oplus M_D \oplus N_G$ using the newly generated nonce $N_G$. It then encrypts the shares as follows $P_U = N_G \oplus s_{G_i} \oplus ID_i$,

$P_D = N_G \oplus s_{G_j} \oplus ID_j$. Further, gateway computes $C_j = h(M||P_D||N_{G_j}||ID_j) \oplus s_{G_j}$ and $C_i = h(M||P_U||N_{G_i}||ID_i) \oplus s_{G_i}$ for verification of the received message at both ends with newly generated $N_{G_j}$ and $N_{G_i}$. Gateway then sends a message $< M, P_D, N_{G_j}, C_j >$ to the device and sends another message $< M, P_U, N_{G_i}, C_i >$ to the user.

*Step $AK3$:* Upon receiving the message $M$, $P_D$, $N_G$, $C_j$, device verifies $N_{G_j}$. Device computes $h(M||P_D||N_{G_j}||ID_j)$ and extracts share of the gateway $s_{G_j} = C_j \oplus h(M||P_D||N_{G_j}||ID_j)$. It verifies the integrity of the message by comparing $C_j \oplus s_{G_j}$ and $h(M||P_D||N_{G_j}||ID_j)$. Further, the device reconstructs the secret using the share of the gateway and its own share as $S_j = (2s_{G_j} - s_j) \ mod \ p$. The device then proceeds to compute $h(S_j||ID_G)$ and compares it with the stored value. At this juncture, the gateway is authenticated to the device. Upon successful authentication, the device computes $V_j = h(S_j||ID_j)$, $N_G = P_D \oplus s_{G_j} \oplus ID_j$, $V_i = M \oplus V_j \oplus N_G$ and the session key $K_S = h(V_i||N_G||V_j)$.

*Step $AK4$:* Upon receiving the message $M$, $P_U$, $N_{G_i}, C_i$, smartphone verifies $N_{G_i}$. Smartphone computes $h(M||P_U||N_{G_i}||ID_i)$ and extracts share of the gateway $s_{G_i} = C_i \oplus h(M||P_U||N_{G_i}||ID_i)$. It verifies the integrity of the message by comparing $C_i \oplus s_{G_i}$ and $h(M||P_U||N_{G_i}||ID_i)$. Further, the smartphone reconstructs the secret using the share of the gateway and its own share as $S_i = (2s_{G_i} - s_i) \ mod \ p$. The device then proceeds to compute $h(S_i||ID_G)$ and compares it with the stored value. At this juncture, the gateway is authenticated to the user. Upon successful authentication, the smartphone computes $V_i = h(S_i||ID_i)$, $N_G = P_U \oplus s_{G_i} \oplus ID_i$, $V_j = M \oplus V_i \oplus N_G$ and the session key $K_S = h(V_i||N_G||V_j)$. Upon successful communication using the newly generated session key, the user and the device are mutually authenticated and also the device is implicitly authenticated to the gateway. Given that the device fails to authenticate, an error message will be sent to the gateway.

The main advantage of this protocol is that only the device identity $ID_j$, identity of the gateway $ID_G$ and its share $s_j$ are stored at the device. An attacker who captures the device learns no knowledge about the keys or the user.

### F. RE-REGISTRATION PHASE

This phase is invoked when the user has lost the smartphone or would like to register a new smartphone. The steps for performing re-registration are as follows:

*Step $RR1$:* User selects the re-registration option in the smart home application and is then prompted to take a face image. The smartphone generates a new nonce $N_i$ and computes $C_i = h(RREG||TD_i||TD_G||N_i||h(I)||K_{U_iG})$ where $TD_i = h(ID_i||K_i||N_i)$ and $TD_G = h(ID_G||K_G||N_i)$. Smartphone sends the following message to the gateway $RREG, TD_i, TD_G, h(I), N_i, C_i, \{I\}_{K_{U_iG}}$.

*Step $RR2$:* Upon reception of the message, the gateway verifies $N_i$ and $C_i$ for checking the integrity of the message. The gateway then decrypts $\{I\}_{K_{U_iG}}$ and verifies the integrity of $I$ with $h(I)$. Further, gateway verifies the freshness of the image. Upon successful verification, the gateway performs face recognition to identify the user. On successful authentication, gateway proceeds to extract $P_{rnu}$ and image information $I_m$ and stores it for future verification.

Further, the gateway computes the secret using the newly generated nonce $N_G$. The gateway and smartphone follow the steps (Step R2 to R3) listed in the registration phase. The user has to follow the authentication and key establishment phase to access a device in the smart home.

## V. FORMAL SECURITY ANALYSIS USING AVISPA

This section presents the formal security analysis using AVISPA and analysis of various security features that are essential to cryptographic protocols.

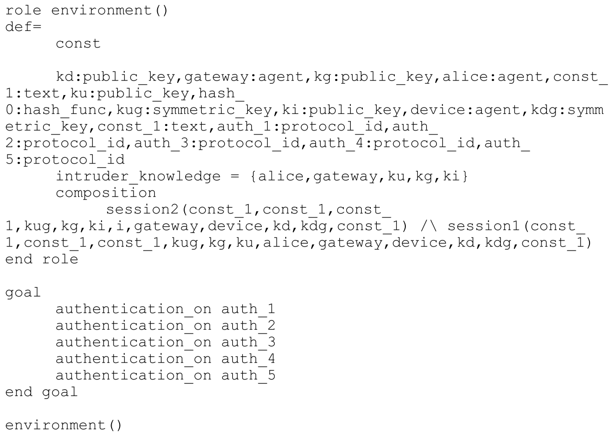
### A. FORMAL SECURITY ANALYSIS USING AVISPA

We use the SPAN+AVISPA (Security Protocol ANimator for Automated Validation of Internet Security Protocols and Applications) tool for performing the formal analysis of the proposed protocol [50] and [51]. Experimental results on many internet security protocols show that the AVISPA tool is state of the art for automated validation of security protocols.

AVISPA uses a High-Level Protocol Specification Language (HLPSL) to represent the cryptographic protocols. The *HLPSL2IF* translator translates the protocol to Intermediate Format (IF) specifications. IF specification is then provided to the back-end modules for analysis. There are mainly four back-ends in the AVISPA tool: which include OFMC, an On-the-fly Model-Checker which detects all known attacks, CL-AtSe, a Constraint-Logic-based Attack Searcher, SATMC, an SAT-based Model-Checker and TA4SP, a Tree Automata based on Automatic Approximations for the Analysis of Security Protocols. AVISPA analyze the protocol under the assumption that the network is under the control of the Dolev-Yao intruder over which is the exchange of messages happens.

We translate both the registration and authentication and key establishment phases of the proposed protocol to HLPSL. The actions of each entity are represented as basic roles. Further, these basic roles are combined to represent the composed role, representing the interactions among them. The entities are represented as user $U$, gateway $G$ and the smart device $D$.

The entities communicate using two different channels: SND and RCV. Finally, an environment role is defined as shown in Fig. 5, which contains global constants and composition of one or more sessions. Besides, it describes the intruder $i$ who plays the role of a legitimate user. The *intruder_knowledge* is specified in the environment session. Finally, the CL-AtSe and OFMC back-ends found the protocol SAFE; in other words, the proposed protocol is secure against the Dolev-Yao threat model used in AVISPA. The implementation also includes the simulation of the intruder attack with the publicly known parameters. The intruder gains no knowledge after capturing the message sent by the user. This shows that the proposed protocol is secure against Man-in-the-middle attacks and replay attacks. The results of the analysis using the CL-AtSe and OFMC back-ends are



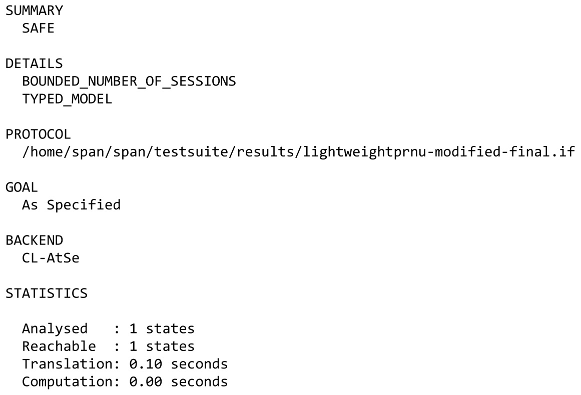
```
role environment()
def=
      const

      kd:public_key,gateway:agent,kg:public_key,alice:agent,const_
1:text,ku:public_key,hash_
0:hash_func,kug:symmetric_key,ki:public_key,device:agent,kdg:symm
etric_key,const_1:text,auth_1:protocol_id,auth_
2:protocol_id,auth_3:protocol_id,auth_4:protocol_id,auth_
5:protocol_id
      intruder_knowledge = {alice,gateway,ku,kg,ki}
      composition
            session2(const_1,const_1,const_
1,kug,kg,ki,i,gateway,device,kd,kdg,const_1) /\ session1(const_
1,const_1,const_1,kug,kg,ku,alice,gateway,device,kd,kdg,const_1)
end role

goal
      authentication_on auth_1
      authentication_on auth_2
      authentication_on auth_3
      authentication_on auth_4
      authentication_on auth_5
end goal

environment()
```

**FIGURE 5.** The environment and goal.



```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/lightweightprnu-modified-final.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 1 states
  Reachable  : 1 states
  Translation: 0.10 seconds
  Computation: 0.00 seconds
```

**FIGURE 6.** AVISPA results using CL-AtSe backend.

depicted in Fig. 6 and Fig. 7 respectively. In other words, the security goals are satisfied by the proposed protocol as specified in the environment.

## B. ANALYSIS OF SECURITY FEATURES

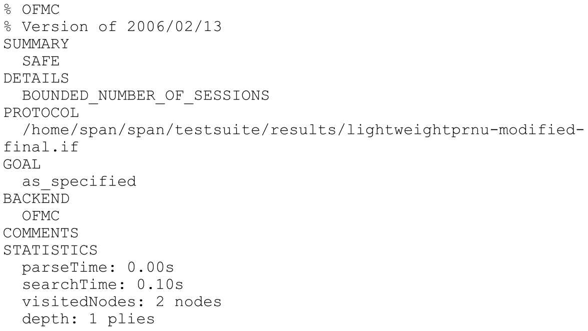
In this section, we analyze the security features of the proposed protocol.

### 1) ANONYMITY

User $U_i$, gateway $G$ and smart device $D_j$ have temporal identities to preserve the anonymity of the communicating entities. They use their public keys for computing the temporal identities. The user, gateway and device compute the temporal identities as $TD_i = h(ID_i||K_{U_i}||N_i)$, $TD_G = h(ID_i||K_G||N_G)$ and $TD_j = h(ID_j)||K_j||N_j)$ respectively. Hence, adversary $A$ who is eavesdropping on the channel will not be able to identify the communicating entities, thus preserving the privacy of all communicating entities.

### 2) KEY FRESHNESS

Key freshness is of paramount importance to a key establishment protocol that ensures that each session's key is randomly generated. In the proposed protocol, to generate a session key both $U_i$ and $D_j$ compute the secrets to obtain $h(S_i||ID_i)$ and $h(S_j||ID_j)$ respectively. Finally, they compute the session key as $K_s = h(h(S_i||ID_i)||N_G||h(S_j||ID_j))$ where $N_G$ is a newly generated nonce for each session. Hence, the freshness of the key is ensured by the presence of nonce generated by $G$, which is a trusted entity.



```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/lightweightprnu-modified-
final.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.10s
  visitedNodes: 2 nodes
  depth: 1 plies
```

**FIGURE 7.** AVISPA results using OFMC backend.

### 3) FORWARD SECRECY

Forward secrecy ensures that the session keys established are not compromised when the long-term key is compromised [52]. Suppose $A$ steals the share $s_j$ of the device $D_j$. $A$ will not be able to determine the secret $S$ as it requires the knowledge of each of the shares and hence will not compute the session keys. Therefore, in this proposed protocol, compromise of any long-term key does not compromise the session keys.

### 4) FAKE GATEWAY ATTACKS

In this attack, $A$ adds a fake gateway to steal the credentials, such as stored keys, or hijacks communication between the user and the smart device. The fake gateway won't be able to decrypt the message $< TD_i, TD_G, TD_j, h(I), N_i, C_i, \{I\}_{K_{U_iG}} >$ as it doesn't own the key $K_{U_iG}$. Moreover, it won't be able to compute the message $< M, P_U, N_{G_i}, C_i >$. Further, if the fake gateway replays a previously sent message $< M, P_U, N_{G_i}, C_i >$, the message will be discarded by the smartphone because of old nonce $N_G$ or unmatched $C_i$ value.

### 5) MAN-IN-THE-MIDDLE ATTACKS

In man-in-the-middle attack (MITM), $A$ intercepts the messages and possibly alters the communications between two entities. Suppose, $A$ relays the message $< TD_i, TD_G, TD_j, h(I), N_A, C_A, \{I\}_{K_{U_iG}} >$ to gateway $G$, $G$ will discard the message upon verification of $N_A$ and $C_A$. Besides, $A$ will not be able to produce a similar response to force user $U_i$ to compute a key which is known to $A$. The key is computed as $K_s = h(V_i||N_G||V_j)$ where $V_i$ and $V_j$ are neither stored at user end nor at the device end. Both entities compute the session key using their corresponding shares. Moreover, suppose $A$ manipulates the content of the message $< M', P_U, N_{G_i}, C_i >$

where $M$ is replaced by $M'$, the proposed protocol will be aborted by the $U_i$ when $h(S_i||ID_G)$ is not matched. Similarly, $A$ will not succeed in establishing a session key with the device by relaying or modifying the message. Hence, the proposed protocol is resilient to MITM attacks.

#### 6) REPLAY ATTACKS

In a replay attack $A$ interferes by replaying a message or a part of a message that was sent previously in any protocol run [53]. Our proposed protocol detects replay attacks through the verification of nonce and integrity. Suppose, $A$ replays the message $< TD_i, TD_G, TD_j, h(I), N_i, C_i, \{I\}_{K_{U_iG}} >$ which was previously sent by the user with a modified $N_i$. $G$ will abort the protocol as $C_i$ won't match with the received value. Moreover, rest of the messages include the nonces in their hash values. Hence, the proposed protocol is resilient to replay attacks.

#### 7) USER IMPERSONATION ATTACKS

Suppose $A$ impersonates as user $U_i$ and sends a photo of user's video. $A$ cannot generate the message $< TD_i, TD_G, TD_j, h(I), N_i, C_i, \{I\}_{K_{U_iG}} >$ as $A$ does not possess the public keys of $U_i$ and $G$ and the pre-shared key $K_{U_iG}$. Hence, the proposed protocol is resilient to user impersonation attacks.

#### 8) SMART DEVICE IMPERSONATION ATTACKS

Suppose $A$ tries to add a device $AD_j$ to the smart home network. $AD_j$ will neither be able to compute the session key as it does not possess a share to reconstruct the secret with $G$ nor be able to compute $h(M||P_D||N_{G_j}||ID_j)$ to extract the share of the gateway as it requires the knowledge of $ID_j$. Hence, the proposed protocol is resilient to smart device impersonation attacks.

#### 9) DENIAL OF SERVICE ATTACKS

There are two types of denial of service attacks (DoS) mainly connection depletion attack and resource depletion attack [54]. Connection depletion attack can be mitigated using local authentication in the smartphone. However, it is difficult to mitigate a resource depletion attack completely. The nonce and hash verification prevent this attack to a great extent. For instance, $A$ can send spurious number of messages to force $G$ to process the message. $G$ aborts execution when a nonce verification fails, or a hash mismatch occurs which prevents it from further processing. Hence, the proposed protocol is resilient to DoS attacks.

#### 10) FINGERPRINT FORGERY ATTACKS

In this attack, $A$ generates a forged image using the $P_{rnu}$ extracted from publicly available images of the user $U_i$. $A$ will not be able to succeed in launching this attack as $A$ does not possess the shared key $K_{U_iG}$, public keys of $U_i$ and $G$, the share $s_i$ and $h(S_i||ID_G)$. Hence, $A$ will not be able to send the forged image to $G$ for authentication. Hence, the proposed protocol is resilient to fingerprint forgery attacks.

#### 11) STOLEN SMART DEVICE ATTACKS

Suppose $A$ obtains physical access to a smart device, A can extract the identities $ID_j, ID_G$, public key $K_D$ and hash of the secret $h(S_j||ID_G)$ using power analysis attacks [55]. $A$ will not be able to compromise the session key as $s_{G_j}$ is also required to compute it. Also, $A$ does not obtain any information regarding the user $U_i$ or any other smart devices in the network. Hence, the proposed protocol is resilient to stolen smart device attacks.

#### 12) SMARTPHONE CAPTURE ATTACKS

In our proposed protocol, we assume that the face recognition scheme is resilient to printed photo attack, printed mask attack and displayed video attack as there are schemes which can prevent these attacks [48]. In that sense, suppose $A$ obtains physical access to the smartphone of the user $U_i$. $A$ will not be able to provide the face biometric of the user $U_i$. Hence, the proposed protocol can withstand stolen smartphone attack.

#### 13) SHOULDER SURFING ATTACKS

Password is used as a common authentication factor in many applications and its ease of use makes the scheme more usable and easy to steal. $A$ can observe directly or use external recording devices to collect user' credentials [18]. We use face recognition and PRNU fingerprint in our proposed protocol to make it resilient to this attack. The PRNU fingerprint verification makes sure that the face image is taken using the user's registered smartphone. According to the proposed protocol, a face image captured using any other camera other than the authenticated smartphone is not accepted as authentic. Hence, the proposed protocol is resilient to shoulder surfing attacks.

## VI. PERFORMANCE EVALUATION

In this section, we present the performance evaluation of the proposed protocol and compare our scheme with existing approaches to demonstrate the lightweight nature of the proposed protocol in terms of both communication and computation overheads. Besides, we show the effectiveness of PRNU by presenting the results of experiments conducted to prove that single reference image is sufficient during registration.

### A. EVALUATION OF PRNU FINGERPRINT

To verify the efficacy of the PRNU fingerprint of smartphones in uniquely identifying devices as proved by Ba *et al.* [56], we use 100 face images collected from more than ten individual smartphones. Smartphones include Vivo 1807, Samsung Galaxy M-30s, Moto G Plus 7015, Redme 8, Vivo V17, Nokia 6.1 Plus, iPhone 7, Realme 2 Pro, Asus Z010D, Realme X, Lenovo A7010a48. We collect the face image captured using the front camera of the smartphones for this purpose. We use the source camera identification algorithm presented in MATLAB source code [57], [58] to test the images. We use Peak Correlation Energy (PCE), which is deemed to be the most used similarity metric, for identifying the source camera or smartphone. PCE is defined as the ratio between the height of the peak and the energy of the cross correlation between reference PRNU and the obtained PRNU patterns [42].

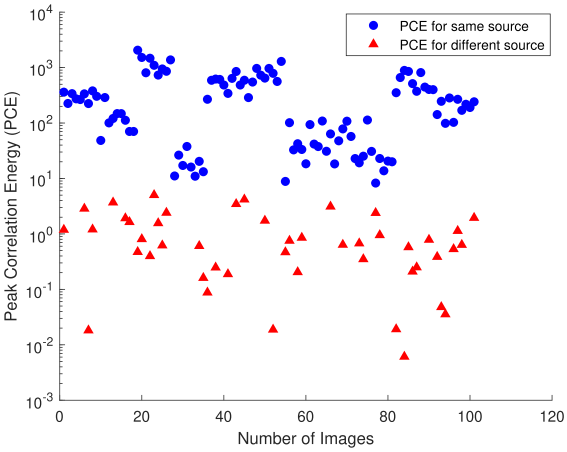


**FIGURE 8.** PCE values when single reference image is used to create the reference PRNU fingerprint.

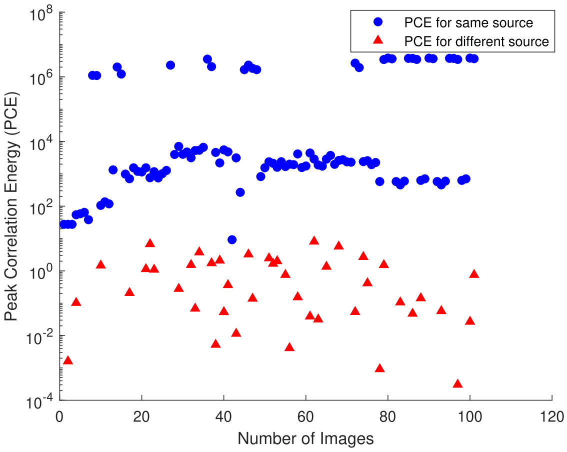


**FIGURE 9.** PCE values when multiple reference images are used to create the reference PRNU fingerprint.

We use MATLAB R2018b [59] for our experiment on a Lenovo/ IBM ThinkPad L480 Laptop running with Windows 10 powered by an Intel Core i5 processor and 4GB of RAM. We compare randomly chosen images with its own set of images as well as images from different sets. Besides, we use multiple reference images and a single reference image for computing the reference fingerprint. The results are depicted in Fig. 8 and Fig. 9 respectively. Our experiments show that in both the cases, the PCE values effectively classify images captured between same and different smartphones based on a threshold. Our results show that though a single image is sufficient to authenticate smartphones, the reference PRNU fingerprint computed using multiple images increases the PCE values. When images are compared against those taken from the same camera, the correlation values are high. On the other hand, we observe the correlation values to be closer to zero when different cameras are used for comparison.

### B. EVALUATION OF THE PROPOSED PROTOCOL

In this section, we present the power and energy analysis of different algorithms used in our proposed protocol. As we use a hybrid approach, we first present the power analysis of various ECC operations such as Elliptic-curve Diffie–Hellman (ECDH), Elliptic Curve Integrated Encryption Scheme (ECIES) and Elliptic Curve Digital Signature Algorithm (ECDSA) [60]. The ECC curves which were selected for analysis include secp112r1, secp128r1 and secp160r1 [61] as these are the commonly used curves. We use Raspberry Pi 3 model B device which is equipped with Quad Core 1.2 GHz Broadcom BCM2837 64 bit CPU with 1 GB RAM, 16 GB SD card and Raspbian Jessie Lite operating system to run these operations. We analyze the power and energy consumption of various ECC operations for a payload of 512 bits of data. The power consumption for each operation is measured using Keysight series B2901A Source Measure Unit [62].

**TABLE 2.** Energy consumption during various ECC operations.

| Operation | secp112r1 (mJ) | secp128r1 (mJ) | secp160r1 (mJ) |
|---|---|---|---|
| EC Key Generation | 0.558 | 0.415 | 0.993 |
| ECIES Encryption | 2.258 | 1.566 | 2.545 |
| ECIES Decryption | 1.577 | 0.949 | 0.837 |
| ECDSA Signature Generation | 0.451 | 0.659 | 0.373 |
| ECDSA Signature Verification | 1.189 | 1.18 | 1.158 |
| ECDH | 0.426 | 0.5 | 1.683 |

We execute Elliptic curve (EC) key generation, ECIES encryption, ECIES decryption, ECDSA signature generation, ECDSA signature verification and ECDH operations and compute the average power and energy consumption. The energy consumption values for each operation are given in Table 2. We infer that the encryption operation consumes more energy than other operations while the signature generation consumes lesser energy than other operations. Moreover, ECC operations are expensive in terms of power and energy consumption than symmetric key operations. We use symmetric key encryption for sending the face image from the user to the gateway. Besides the protocol uses temporal identities which are unique for each session and hence, the protocol satisfies anonymity and untraceability properties.

Further, we implement image hash, encryption and decryption in Python using AES (Advanced Encryption Standard) algorithm for various sizes of images using Raspberry Pi. The execution time and energy consumption for these operations are listed in Table 3. We infer that as the size of the face image increases the execution time increases and hence the energy consumption. Therefore, it is desirable to use a smaller size image for user authentication.

**TABLE 3. Execution time and energy consumption of various operations on the face images.**

| Image size | Hash Time | Encryption Time (ms) | Encryption E ($\mu$J) | Decryption Time (ms) | Decryption E ($\mu$J) |
|---|---|---|---|---|---|
| 64x85 | 0.3 | 4.15 | 75.5 | 4.2 | 80.5 |
| 250x250 | 0.54 | 16.4 | 325 | 15.9 | 312 |
| 400x640 | 1.03 | 55.91 | 1085 | 47.8 | 911 |

## C. PERFORMANCE COMPARISON

### 1) SECURITY FEATURES COMPARISON

The comparison of security features such as mutual authentication, anonymity and resilience to various attacks are listed in Table 4.

Wazid *et al.* [21] proposed an efficient lightweight authentication protocol. However, their informal security analysis section states that an adversary can obtain the session key if he captures the device. Thus, the compromise of the session keys can reveal the messages exchanged between the user and the device. Hence, the scheme does not provide security against smart device capture attacks. Similarly, the scheme of Challa *et al.* [36] is also vulnerable to smart device capture attack as it can lead to the compromise of the session key. Besides, Chaudhry *et al.* [63] pointed out the correctness issues of this scheme and argued that it cannot complete operations normally.

The scheme proposed by Yu *et al.* [40] does not provide integrity protection to verify whether messages are modified in transit. Hence, entities have to perform computations to authenticate the message. Consequently, their scheme is vulnerable to DoS attacks.

The scheme proposed by Shuai *et al.* [24] is vulnerable to replay attacks due to the lack of nonce or timestamp verification. Besides, the scheme is vulnerable to user impersonation attacks and shoulder surfing attacks as the method is based on simple password authentication. Moreover, the scheme doesn't provide anonymity as the identities of the user and the device are always encrypted using the same master key.

In contrast to the existing approaches, our scheme provides features such as anonymity, key freshness, session key establishment, resilience to smart device capture attacks and fake gateway attacks.

### 2) COMPARISON OF COMMUNICATION OVERHEAD

We make a general assumption that the length of the identities of the user, gateway and the smart device, randomly generated nonces, timestamps and the message digest are 128, 128, 64, 128, 77 (verified in MATLAB) and 160 bits respectively.

Based on the above assumption, we compute the communication overhead at the device as it is resource-constrained, while $G$ and $U_i$ are assumed to be resource-rich. The communication cost of our scheme for the message sent from the gateway to the smart device is 544 bits (68 bytes) ($128 + 128 + 128 + 160 = 544$). Similarly, the communication costs of the existing protocols proposed by Wazid *et al.* [21] is 986 bits, Yu *et al.* [40] is 794 bits, Challa *et al.* [36] is 711 bits and Shuai *et al.* [24] is 960 bits. Table 5 shows the communication overhead at the smart device, which includes the number of messages received and sent and the total cost in bits. Our proposed protocol uses the lowest number of messages and the smallest size of messages compared to the other protocols, which are practical in such smart home applications. Hence, it implies that the proposed protocol consumes lesser power compared to the other schemes.

**TABLE 4. Security features comparison with existing schemes.**

| Security features | [21] | [40] | [36] | [24] | Our |
|---|---|---|---|---|---|
| Anonymity | Y | Y | Y | N | Y |
| Key Freshness | Y | Y | Y | Y | Y |
| Key Establishment | Y | Y | Y | Y | Y |
| Mutual Authentication | Y | Y | Y | Y | Y |
| Forward Secrecy | Y | Y | Y | Y | Y |
| Resilience to replay attack | Y | Y | Y | N | Y |
| Resilience to user impersonation attack | Y | Y | Y | N | Y |
| Resilience to device impersonation attack | Y | Y | Y | Y | Y |
| Resilience to MITM attack | Y | Y | Y | Y | Y |
| Resilience to DoS attack | Y | N | Y | Y | Y |
| Resilience to smartphone capture attack | Y | Y | Y | Y | Y |
| Resilience to device capture attack | N | Y | N | Y | Y |
| Resilience to shoulder surfing attack | Y | Y | Y | N | Y |

**TABLE 5. Communication overhead comparison with existing schemes.**

| Scheme | Number of messages | Total Cost (bits) |
|---|---|---|
| Wazid *et al.* [21] | 2 | 986 |
| Yu *et al.* [40] | 2 | 794 |
| Challa *et al.* [36] | 2 | 711 |
| Shuai *et al.* [24] | 2 | 960 |
| Proposed | 1 | 544 |

### 3) COMPARISON OF COMPUTATION OVERHEAD

Based on the approximate time [64] given in Table 6, we compute the computation overhead for each scheme. The computation overhead at the smart device incurred by each scheme is given in Table 7. The computation cost for the proposed scheme is estimated to be $1.28ms$ which comprises the computation cost for 4 hash operations and a negligible cost for secret reconstruction when tested in Raspberry Pi ($4T_H \approx 0.00128 + 0.00000405$).

The comparison of communication and computation costs with existing schemes in terms of a number of message exchanges and approximate computation time at the smart device is depicted in Fig. 10. The proposed scheme incurs a significant reduction in the number of message exchanges, computation costs and satisfies security features. Even though the computation cost is negligibly higher than [24], the security aspects of the proposed scheme outperform other schemes. Hence, we can conclude that the proposed protocol is lightweight and can be applied to smart home environments.

**TABLE 6. Approximate time required for cyptographic operations.**

| Notation | Operation | Approximate Time (s) |
|---|---|---|
| $T_H$ | Hash Function | 0.00032 |
| $T_S$ | Symmetric key operations | 0.0056 |
| $T_{MI}$ | Modular inverse | 0.00004275 |
| $T_M$ | Modular multiplication | 0.0001425 |
| $T_{PM}$ | Point Multiplication | 0.0171 |

**TABLE 7. Computation overhead comparison with existing schemes.**

| Schemes | Total cost | Estimated Time (ms) |
|---|---|---|
| [21] | $7T_H + T_S$ | 7.84 |
| [40] | $7T_H$ | 2.24 |
| [36] | $5T_H$ | 1.6 |
| [24] | $3T_H$ | 0.96 |
| Proposed | $4T_H$ | 1.28 |

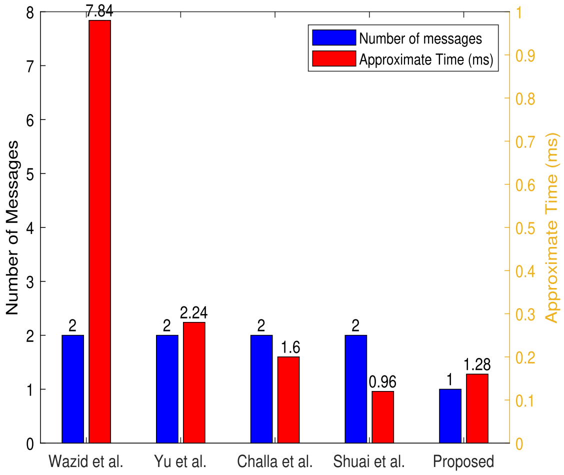


**FIGURE 10. Communication and computation cost comparison with existing schemes.**

## VII. DISCUSSION

Although our proposed protocol has been prototyped on embedded devices, it needs to be evaluated in realistic smart home environments to understand the behavioral characteristics of the protocol from usability and security perspectives. Performance and energy consumption need to be estimated in an end-to-end manner. For instance, cryptographic operations and remote face recognition may incur processing and transmission delays respectively. Similarly, remote face recognition needs to consider the aging and illumination conditions of diverse users.

## VIII. CONCLUSION

In this paper, we proposed a lightweight and privacy-preserving remote user authentication protocol using geometric secret sharing for smart home environments. The proposed protocol is designed to avoid the use of passwords and smart cards and hence alleviates the burden of users in carrying additional mechanisms and in memorizing long passwords. Performance analysis including power consumption and comparison with other existing schemes revealed that the proposed protocol is lightweight, privacy-preserving, usable and prevents attacks specifically phishing and fake gateway and smart device impersonation attacks. Further, we conducted experiments to show the effectiveness of PRNU in identifying the smartphone of the users. Moreover, the formal security analysis using AVISPA and performance evaluation using Raspberry Pi showed that the proposed protocol is highly secure to provide enhanced security to smart homes.

As part of future work, we plan to implement this protocol in a real-time smart home environment. In addition, we propose to conduct user studies to evaluate the effectiveness of the authentication protocol for diverse environmental conditions.

## REFERENCES

[1] ERICSSON. *Internet of Things forecast*. Accessed: May 2, 2019. [Online]. Available: https://www.ericsson.com/en/mobility-report/internet-of-things-forecast

[2] E. Fernandes, J. Jung, and A. Prakash, "Security analysis of emerging smart home applications," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2016, pp. 636–654.

[3] N. Lomas. *Critical Flaw IDed in ZigBee Smart Home Devices*. Accessed: May 9, 2018. [Online]. Available: https://techcrunch.com/2015/08/07/critical-flaw-ided-in-zigbee-smart-home-devices/

[4] S. G. B. Fouladi. *Honey, I'm Home!! Hacking Z-Wave Home Automation Systems*. Accessed: May 9, 2018. [Online]. Available: https://cybergibbons.com/wp-content/uploads/2014/11/honeyimhome-1310010% 42426-phpapp01.pdf

[5] E. Y. Chen, Y. Pei, S. Chen, Y. Tian, R. Kotcher, and P. Tague, "OAuth demystified for mobile application developers," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Nov. 2014, pp. 892–903.

[6] A. Lakhani. (2020). *Understanding the Impact of COVID-19 on IoT Security*. Accessed: Jan. 2021. [Online]. Available: https://www.fortinet.com/blog/industry-trends/understanding-the-impact-of-covid-19-on-iot-security

[7] E. K. Parsons, E. Panaousis, and G. Loukas, "How secure is home: Assessing human susceptibility to IoT threats," in *Proc. 24th Pan-Hellenic Conf. Informat.*, Nov. 2020, pp. 64–71.

[8] J. Sengupta, S. Ruj, and S. Das Bit, "A comprehensive survey on attacks, security issues and blockchain solutions for IoT and IIoT," *J. Netw. Comput. Appl.*, vol. 149, Jan. 2020, Art. no. 102481.

[9] S. Debroy, P. Calyam, M. Nguyen, R. L. Neupane, B. Mukherjee, A. K. Eeralla, and K. Salah, "Frequency-minimal utility-maximal moving target defense against DDoS in SDN-based systems," *IEEE Trans. Netw. Service Manage.*, vol. 17, no. 2, pp. 890–903, Jun. 2020.

[10] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "DDoS in the IoT: Mirai and other Botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017.

[11] V. Clincy and H. Shahriar, "IoT malware analysis," in *Proc. IEEE 43rd Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, Jul. 2019, pp. 920–921.

[12] M. A. Khan and K. Salah, "IoT security: Review, blockchain solutions, and open challenges," *Future Gener. Comput. Syst.*, vol. 82, pp. 395–411, May 2018.

[13] M. Chen, J. Fridrich, M. Goljan, and J. Lukáš, "Source digital camcorder identification using sensor photo response non-uniformity," *Proc. SPIE*, vol. 6505, Feb. 2007, Art. no. 65051G.

[14] S. Jia, C. Hu, X. Li, and Z. Xu, "Face spoofing detection under super-realistic 3D wax face attacks," *Pattern Recognit. Lett.*, vol. 145, pp. 103–109, May 2021.

[15] Z. Ba, S. Piao, X. Fu, D. Koutsonikolas, A. Mohaisen, and K. Ren, "ABC: Enabling smartphone authentication with built-in camera," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2018, pp. 18–21.

[16] B. Ali and A. Awad, "Cyber and physical security vulnerability assessment for IoT-based smart Homes," *Sensors*, vol. 18, no. 3, p. 817, 2018.

[17] L. D. Tsobdjou, S. Pierre, and A. Quintero, "A new mutual authentication and key agreement protocol for mobile client—Server environment," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 2, pp. 1275–1286, Jun. 2021.

[18] H.-M. Sun, S.-T. Chen, J.-H. Yeh, and C.-Y. Cheng, "A shoulder surfing resistant graphical authentication system," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 2, pp. 180–193, Mar./Apr. 2018.

[19] K. Zheng, T. Wu, X. Wang, B. Wu, and C. Wu, "A session and dialogue-based social engineering framework," *IEEE Access*, vol. 7, pp. 67781–67794, 2019.

[20] Z. Ba and K. Ren, "Addressing smartphone-based multi-factor authentication via hardware-rooted technologies," in *Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2017, pp. 1910–1914.

[21] 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, Mar. 2020.

[22] 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. 1–23, 2019.

[23] D. Kaur and D. Kumar, "Cryptanalysis and improvement of a two-factor user authentication scheme for smart home," *J. Inf. Secur. Appl.*, vol. 58, May 2021, Art. no. 102787.

[24] 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.

[25] J. Patman, D. Chemodanov, P. Calyam, K. Palaniappan, C. Sterle, and M. Boccia, "Predictive cyber foraging for visual cloud computing in large-scale IoT systems," *IEEE Trans. Netw. Service Manage.*, vol. 17, no. 4, pp. 2380–2395, Dec. 2020.

[26] T. Song, R. Li, B. Mei, J. Yu, X. Xing, and X. Cheng, "A privacy preserving communication protocol for IoT applications in smart Homes," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1844–1852, Dec. 2017.

[27] A. Lakhani. (Nov. 2020). *Understanding the Impact of COVID-19 on IoT Security*. [Online]. Available: https://www.fortinet.com/blog/industry-trends/understanding-the-impact-of-covid-19-on-iot-security

[28] G. P. W. Whitmore. (Apr. 2020). *COVID-19 Cyberwar: How to Protect Your Business*. [Online]. Available: https://www.ibm.com/downloads/cas/Y5QGA7VZ

[29] B. Pranggono and A. Arabo, "COVID-19 pandemic cybersecurity issues," *Internet Technol. Lett.*, vol. 4, no. 2, p. e247, 2021.

[30] M. Grini. *Coronavirus: Working Remotely Requires Increased Security*. (Mar. 2020). [Online]. Available: https://www.ibm.com/blogs/nordic-msp/covid-19-remote-work-security/

[31] R. O. Andrade, I. Ortiz-Garces, and M. Cazares, "Cybersecurity attacks on smart home during COVID-19 pandemic," in *Proc. 4th World Conf. Smart Trends Syst., Secur. Sustainability (WorldS)*, Jul. 2020, pp. 398–404.

[32] P. Kumar, A. Gurtov, J. Iinatti, M. Ylianttila, and M. Sain, "Lightweight and secure session-key establishment scheme in smart home environments," *IEEE Sensors J.*, vol. 16, no. 1, pp. 254–264, Jan. 2015.

[33] P. Kumar, A. Braeken, A. Gurtov, J. Iinatti, and P. H. Ha, "Anonymous secure framework in connected smart home environments," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 4, pp. 968–979, Apr. 2017.

[34] 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.

[35] M. Fakroon, M. Alshahrani, F. Gebali, and I. Traore, "Secure remote anonymous user authentication scheme for smart home environment," *Internet Things*, vol. 9, Mar. 2020, Art. no. 100158.

[36] S. Challa, A. K. Das, V. Odelu, N. Kumar, S. Kumari, M. K. Khan, and A. V. Vasilakos, "An efficient ECC-based provably secure three-factor user authentication and key agreement protocol for wireless healthcare sensor networks," *Comput. Elect. Eng.*, vol. 69, pp. 534–554, Jul. 2018.

[37] K. Nimmy, S. Sankaran, and K. Achuthan, "A novel multi-factor authentication protocol for smart home environments," in *Information Systems Security*, vol. 11281, V. Ganapathy, T. Jaeger, R. Shyamasundar, Eds. Cham, Switzerland: Springer, 2018, pp. 44–63.

[38] B. Vaidya, S.-S. Yeo, J. J. P. C. Rodrigues, and J. H. Park, "Robust one-time password authentication scheme using smart card for home network environment," *IEEE Comput. Commun.*, vol. 34, no. 3, pp. 326–336, Mar. 2011.

[39] S. Jegadeesan, M. Azees, N. Ramesh Babu, U. Subramaniam, and J. D. Almakhles, "EPAW: Efficient privacy preserving anonymous mutual authentication scheme for wireless body area networks (WBANs)," *IEEE Access*, vol. 8, pp. 48576–48586, 2020.

[40] S. Yu, N. Jho, and Y. Park, "Lightweight three-factor-based privacy-preserving authentication scheme for IoT-enabled smart Homes," *IEEE Access*, vol. 9, pp. 126186–126197, 2021.

[41] S. K. Mani, R. Durairajan, P. Barford, and J. Sommers, "An architecture for IoT clock synchronization," in *Proc. 8th Int. Conf. Internet Things*, 2018, pp. 1–8.

[42] M. Goljan, J. Fridrich, and M. Chen, "Defending against fingerprint-copy attack in sensor-based camera identification," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 1, pp. 227–236, Mar. 2011.

[43] J. Lukäš, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 2, pp. 205–214, Jun. 2006.

[44] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979.

[45] G. Blakley and G. Kabatianskii, "Linear algebra approach to secret sharing schemes," in *Proc. Workshop Inf. Protection*. Berlin, Germany: Springer, 1993, pp. 33–40.

[46] A. Shamsoshoara, "Overview of Blakley's secret sharing scheme," 2019, *arXiv:1901.02802*.

[47] D. Dolev and A. C. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 2, pp. 198–208, Mar. 1983.

[48] D. Wen, H. Han, and A. K. Jain, "Face spoof detection with image distortion analysis," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 4, pp. 746–761, Apr. 2015.

[49] T. Cooijmans, J. de Ruiter, and E. Poll, "Analysis of secure key storage solutions on android," in *Proc. 4th ACM Workshop Secur. Privacy Smartphones Mobile Devices*, 2014, pp. 11–20.

[50] Y. Glouche, T. Genet, and E. Houssay, "SPAN: A security protocol animator for AVISPA—User manual," IRISA/Univ. Rennes 1, Tech. Rep., 2006.

[51] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuéllar, and P. H. Drielsma, "The AVISPA tool for the automated validation of internet security protocols and applications," in *Proc. Int. Conf. Comput. Aided Verification*. Springer, 2005, pp. 281–285.

[52] M. Azees, P. Vijayakumar, M. Karuppiah, and A. Nayyar, "An efficient anonymous authentication and confidentiality preservation schemes for secure communications in wireless body area networks," *Wireless Netw.*, vol. 27, no. 3, pp. 2119–2130, Apr. 2021.

[53] P. Vijayakumar, M. S. Obaidat, M. Azees, S. H. Islam, and N. Kumar, "Efficient and secure anonymous authentication with location privacy for IoT-based WBANs," *IEEE Trans. Ind. Informat.*, vol. 16, no. 4, pp. 2603–2611, Apr. 2020.

[54] C. Boyd and A. Mathuria, *Protocols for Authentication and Key Establishment*. Germany: Springer-Verlag, 2013.

[55] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Trans. Comput.*, vol. 51, no. 5, pp. 541–552, May 2002.

[56] Z. Ba, S. Piao, X. Fu, D. Koutsonikolas, A. Mohaisen, and K. Ren, "ABC: Enabling smartphone authentication with built-in camera," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2018, pp. 1–5.

[57] M. Chen, J. Fridrich, M. Goljan, and J. Lukás, "Determining image origin and integrity using sensor noise," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 1, pp. 74–90, Mar. 2008.

[58] M. Goljan, M. Chen, P. Comesaña, and J. Fridrich, "Effect of compression on sensor-fingerprint based camera identification," *Electron. Imag.*, vol. 2016, no. 8, pp. 1–10, Feb. 2016.

[59] MathWorks. (2018). *MATLAB R2018b*. [Online]. Available: https://www.mathworks.com/content/dam/mathworks/mathworks-dot-com/support/updates/r2018b/r2018b-updates-release-notes.pdf

[60] D. Hankerson, A. J. Menezes, and S. Vanstone, *Guide to Elliptic Curve Cryptography*. New York, NY, USA: Springer-Verlag, 2006.

[61] C. Research, *SEC 2: Recommended Elliptic Curve Domain Parameters*, accessed on 2018-01-09. [Online]. Available: https://www.secg.org/sec1-v2.pdf

[62] K. Technologies. (2020). *B2900 Series Precision Source/Measure Units (SMU)*. [Online]. Available: https://www.keysight.com/in/en/products/source-measure-units-smu/b2900-series-precision-source-measure-units-smu.html

[63] S. A. Chaudhry, T. Shon, F. Al-Turjman, and M. H. Alsharif, "Correcting design flaws: An improved and cloud assisted key agreement scheme in cyber physical systems," *Comput. Commun.*, vol. 153, pp. 527–537, Mar. 2020.

[64] D. He, N. Kumar, J.-H. Lee, and R. S. Sherratt, "Enhanced three-factor security protocol for consumer USB mass storage devices," *IEEE Trans. Consum. Electron.*, vol. 60, no. 1, pp. 30–37, Feb. 2014.

**K. NIMMY** (Student Member, IEEE) received the Bachelor of Technology (B.Tech.) degree in computer science and engineering from the Rajiv Gandhi Institute of Technology, Kerala, and the Master of Technology (M.Tech.) degree in cyber security from the TIFAC CORE in Cyber Security, Amrita Vishwa Vidyapeetham (Amrita University), Coimbatore. She is a Ph.D. Scholar at the Center for Cyber Security Systems and Networks, Amrita Vishwa Vidyapeetham. Prior to that, she was working as an Assistant Professor in a reputed engineering college. She has more than four years of experience in research projects funded by the Indian Space Research Organization (ISRO) and more than three years of teaching experience. Her research interests include authentication protocols, the IoT security, web security, and cryptography.

**SRIRAM SANKARAN** (Member, IEEE) received the B.Tech. degree (Hons.) from the Malaviya National Institute of Technology Jaipur, Jaipur, formerly known as the Regional Engineering College, and the M.S. and Ph.D. degrees from the University at Buffalo, The State University of New York. He is an Assistant Professor with the Center for Cybersecurity Systems and Networks, Amrita Vishwa Vidyapeetham, Amritapuri Campus, where he directs the Sustainable Computing Laboratory. He has published more than 50 papers all in the areas of mobile embedded computing systems and cybersecurity. His research interests include mobile embedded computing systems, cybersecurity, and the Internet of Things, with a particular focus on energy efficient computing, modeling, and simulation. He served as the Program Chair for the International Symposium on Embedded Computing and System Design (ISED) 2019 and also bagged the Best Paper Award at the IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS) 2020.

**KRISHNASHREE ACHUTHAN** (Senior Member, IEEE) received the Ph.D. degree from Clarkson University, NY, USA. She currently heads the Center for Cybersecurity Systems and Networks and the Amrita Technology Business Incubator (Amrita TBI), Amrita Vishwa Vidyapeetham. She is also the Dean of P.G. Programs at the Amrita School of Engineering, Coimbatore. She is an Ardent Researcher with multi-disciplinary interests. She holds 29 U.S. patents and has published widely in highly acclaimed international journals.

**PRASAD CALYAM** (Senior Member, IEEE) received the M.S. and Ph.D. degrees from the Department of Electrical and Computer Engineering, The Ohio State University, in 2002 and 2007, respectively. He is currently an Associate Professor with the Department of Computer Science, University of Missouri, Columbia. Previously, he was the Research Director at the Ohio Supercomputer Center. His research interests include distributed and cloud computing, computer networking, and cyber security.