ISSN: 1803-3814 (Printed), 2571-3701 (Online)
https://doi.org/10.13164/mendel.2023.2.181

An Integrated Two-Factor Authentication Scheme for Smart Communications and Control Systems

Hoang Trong Minh[1], Bui Van Hau[2], and Nguyen Nam Hoang[3,✉]

[1]Telecommunication faculty 1, Posts and Telecommunications Institute of Technology, Hanoi, 100000, Vietnam
[2]Electronic Faculty, The University of Econ-Tech for Industries, 456 Minh Khai Street, Hanoi, 100000, Vietnam
[3]University of Engineering and Technology, Vietnam National University, Hanoi, 100000, Vietnam
hoangnn@vnu.edu.vn✉, hoangtrongminh@ptit.edu.vn and bvhau@uneti.edu.vn

Abstract

Fast and reliable authentication is a crucial requirement of communications networks and has various research challenges in an Internet of Things (IoT) environment. In IoT-based applications, as fast and user-friendly access and high security are required simultaneously, biometric identification of the user, such as the face, iris, or fingerprint, is broadly employed as an authentication approach. Moreover, a so-called multi-factor authentication that combines user identification with other identification information, including token information and device identity, is used to enhance the authentication security level. This paper proposes a novel two-factor authentication scheme for intelligent communication and control systems by utilizing the watermarking technique to incorporate the mobile device authentication component into the user's facial recognition image. Our proposed scheme offers user-friendliness while improving user security and privacy and reducing authentication information exchange procedures to provide a secure and lightweight schema in real applications. The proposed scheme's security advantages are validated using the widely accepted Burrows–Abadi–Needham (BAN) logic and experimentally assessed using the Automated Validation of Internet Security Protocols and Applications (AVISPA) simulator tool. Finally, our experimental results show that the proposed authentication scheme is an innovative solution for a smart-home control system, such as a smart lock door operation.

Keywords: *Security, Authentication, Watermarking Technique, Face Recognition, Smart-home.*

Received: 22 August 2023
Accepted: 27 September 2023
Online: 18 October 2023
Published: 20 December 2023

1 Introduction

The number of communications devices connecting to the Internet has rapidly increased, facilitating various practical applications in commercial, industrial, and personal application scenarios. While the massive deployment of IoT devices changes how people run their businesses and daily lives, unauthorized access from IoT devices to IoT systems imposes some serious security threats. Smart-home management applications via smartphones have been widely renowned recently because of their convenience. However, since the data between a user device (e.g., a smartphone) and the smart home's gateway is often transmitted over insecure wireless communications links, various cyberattacks, namely user impersonation, device ID clones, and modification, can arise in the smart home environment, [2], [13]. Secure authentication is one of the most critical security functions that should be prioritized to be developed at the highest security level. Conventional authentication schemes were previously designed to exploit only one authentication factor—either one of the personal belongings (e.g., IC cards, tokens, and keys) or information about the user's identity (e.g., a PIN, a password). Modern multi-factor authentication schemes have been engineered by exploiting biometric information such as fingerprints, retina scans, facial/voice recognition, and other information to strengthen the system's security level. However, security, convenience, performance, and application deployment remain the foremost concerns of academics and developers.

The multi-factor authentication scheme is viewed as an end-user identity supplement. Adding an authentication element improves the level of security, but it can also be inconvenient for the user or increase the workload associated with processing sensitive data. Not only do smart device application services have limited processing capability, but they are also frequently strongly related to human behavior. Therefore, developing a user-friendly, secure, and effective authentication procedure is vital.

To prioritize the convenience of smart door opening applications in smart homes, we offer a transparent mechanism for two-factor authentication that combines facial recognition and mobile device identification. Accordingly, the second authentication element is gener-

ated automatically without requiring additional action from the user or device. Furthermore, a watermarking approach is applied to the user's image transferred from the user's device and the gateway to simplify the authentication procedures and protect the user's privacy from attacks on the wireless transmission in an open environment. Our proposal has been assessed for security using BAN logic, validated for safety using a formal model as Automated Validation of Internet Security-sensitive Protocols and Applications (AVISPA), and compared to prior research to demonstrate its merits and applicability.
This study aims to enhance the security level for intelligent communication and control systems by combining the authentication component of mobile devices into the user's facial image recognition with the watermarking technique. The main contributions are summarized as follows:

- A multi-factor authentication scheme based on facial recognition and mobile device ID is proposed to enhance user convenience and secure smart home access.
- By utilizing the watermarking technique to ease the procedure by embedding the session key in the user image and simultaneously protecting the user image's privacy.
- The proposed scheme is verified by the AVISPA testing toolkit or logic analysis and evaluated in the real application setting to validate the feasible implementation and its security impact.

The paper is structured as follows: the next section summarizes the research-related work of recent proposals. Session III focused on our proposed schema and its novel processes. Session IV will include an evaluation of the proposed schema and security analysis, as well as a comparison of its functionality to that of previous schemes. Session V presents an example of a prototype evaluation to demonstrate the practicality of the concept. Session VI will conclude with a discussion of the pros and disadvantages of the schema and the direction of future development.

2 Related Work

Multi-factor authentication is of great interest to researchers because it improves access security, especially for communication applications in an open environment. A common approach is that besides bio-metric user authentication, a second authentication factor is supplemented by what the user has or knows. This second factor can either be effective or transparent to the user.

The authors in [17] have applied combined facial recognition and Radio-frequency identification (RFID) to increase the authentication accuracy for smart home access service users. The system's authentication performance has high accuracy, and the access time meets the requirements of smart home door opening (under 10s). However, this study needs to add an RFID identification device and can only be evaluated experimentally. Therefore, the convenience has been reduced, and the logic of the safety evaluation of the scheme is incomplete. Taking the same approach to device addition, the authors in [20] have proposed using a card reader instead of RFID. The proposed scheme has been security analyzed through BAN logic and proven resistant to side attacks, but this proposal still requires additional user validation action. In [7], develop a door lock system based on facial recognition with two-factor authentication using OpenCV. The design of this project is based on human face recognition and a One Time Password (OTP) solution using the Twilio service. Despite achieving high security, the system requires a communication solution from a 3rd party.

In order to have the most convenient access for users and integrate authentication components to meet the requirements of lightweight authentication procedures, various authors have proposed several two-factor authentication schemes for smart homes as below. A scheme called TFA (Transparent Two-Factor Authentication) is proposed in [22] to avoid tedious interaction and bring satisfaction in the user experience by integrating two authentication components in one User action. Specifically, the voiceprint method is used as the second authentication factor. However, with AI's strong development in facial recognition, the facial recognition solution is accurate and effective. in smart home access practice [7] [12] [21]. Facial recognition will provide an efficient user experience in applications close to everyday human interaction. In addition, these recommendations emphasize user-friendliness and empirical modeling. However, logical analyses to verify the proposal are not provided.

Considering the smart home application as an IoT application, [10] offers a two-factor authentication solution for smart homes using elliptic curve cryptography (ECC) system to resist phone calls security attacks, including impersonation attacks and session key disclosure while ensuring secure user authentication. It is suggested to use fuzzy extraction to improve the security of the two-factor authentication scheme and ensure efficient performance because it uses only the hash function, and XOR operation generates low computational cost. However, the user still needs to verify the identity through the password, which increases the user's complexity and is not safe from guesting password attacks. In addition, the Schema is only evaluated for security through the BAN logic and the formal model Proveif. There are no media modeling assessments like AVISPA. Along with the solution of using ECC curve encryption and random number matching in session communication, the author in [14] proposed a performance and security balance scheme asymptotic to the real environment. However, some inside attacks or session locking are not guaranteed. To deal with the flaws of the above proposal.

In [18], a proposed authentication scheme for a remote access solution to a smart home. Which uses one-way hash functions, bit XOR operations, and symmetric encryption/decryption. The security of the proposal is proven through the Real-Or-Random (ROR) model and the security verification by using the tool (AVISPA). However, this solution uses bio-metric and password and is not completely transparent (transparency), causing user problems. Sensor networks serving the medical field have higher security and authentication needs than conventional networks because they involve sensitive user data. In [19] presented a two-factor authentication mechanism based on a secret and shared key between the gateway and sensor. In [15] propose a resilient ECC-based three-factor mutual authentication protocol with key establishment technique. However, increasing the key length or using public key techniques burdens the processing and slows the authentication time.

During media transmission and storage, data can be altered for illegal use by attackers. Watermarking effectively protects vulnerable data in a digital environment against the tampering of intellectual property rights and enhances security [11]. The authors in [9] proposed a watermarking algorithm based on a lossy compression algorithm to ensure authentication and forgery detection. A cryptography-based bit-pair matching watermarking mechanism in the spatial domain was suggested in [4], where symmetric key cryptography was used for watermark encryption to protect information from intruders on the communication channel. The watermarking mechanism can improve security while minimizing the growth of the security traffic. To avoid exposing embedded bits of an image to attackers, the authors in [3] proposed a block-based image watermarking algorithm. The algorithm generates two different keys using Diffie-Hellman Key Exchange to find the position of the cover image to which watermark bits are to be embedded. The above proposals show that the applicability of the watermarking technique in user image data transmission reduces the amount of information to be transmitted, reduces the examined procedures, and can be used to transmit authentication information. However, previous studies have not mentioned solutions to protect the integrity of image data and use it for session key transmission.

Based on a review of the aforementioned studies and to the best of our knowledge, a 2-factor authentication scheme that supports user convenience, a lightweight protocol, and an increased security level is not fully addressed. The watermarking technique was described in our previous study [8] as a method for embedding a random key generated from the device address into the user image. However, this scheme did not encrypt the user's outgoing messages or perform performance test assessments. To improve the security and privacy of the user's image, this proposal proposes encryption of the entire outgoing message with a session key validated by both the user's device and the gateway. In addition to security analysis with BAN logic, this work continues to expand security verification utilizing the AVISPA tool and practical experiments.

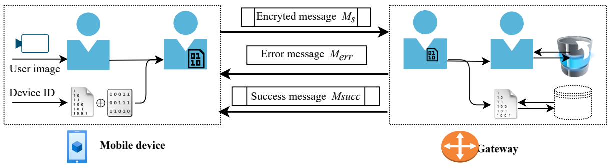


Figure 1: The model of the two-factor authentication system.

This paper presents an authentication solution that effectively integrates two identity factors into an explicit schema with the support of standard security algorithms, such as hash function, Bit-wise XOR, and symmetric cryptography. The proposed scheme can enhance security by using randomly generated session keys and offer a user-friendly and lightweight procedure based on face recognition over a single authentication message. The logic BAN analysis and the AVISPA tool have examined the security analysis of the proposed scheme. We also built an experimental testbed and proved that the response time of the proposed security procedure meets the user requirements for smart-home applications.

3 System Model and the Proposed Authentication Scheme

3.1 System Model and the Two-factor Authentication Diagram

Figure 1 shows the proposed two-factor authentication system model consisting of a user's mobile device (smartphone) and a gateway for illustration. The system deploys the two-factor authentication scheme that composes a user's face captured by the smartphone's camera and the hardware identifier of the user's mobile device. Facial recognition authentication ensures the validity of the user's access to the system when it matches the database available at the gateway. This user-friendly approach allows users to access the system easily but at the risk of counterfeiting, cloning, or privacy violations. To avoid such risks, a mobile device identifier is dynamically employed for each connection session and is used as the session key to protect the user image information transmitted between the user and the gateway. The encrypted user, image, and authentication information are embedded in a single encrypted message for participant exchange using watermarking techniques. As a result, the number of authentication exchange procedures and the data packet size are significantly reduced, and a lightweight authentication protocol is obtained.

Figure 2 shows that the two-factor authentication diagram includes the setup and authentication phases.

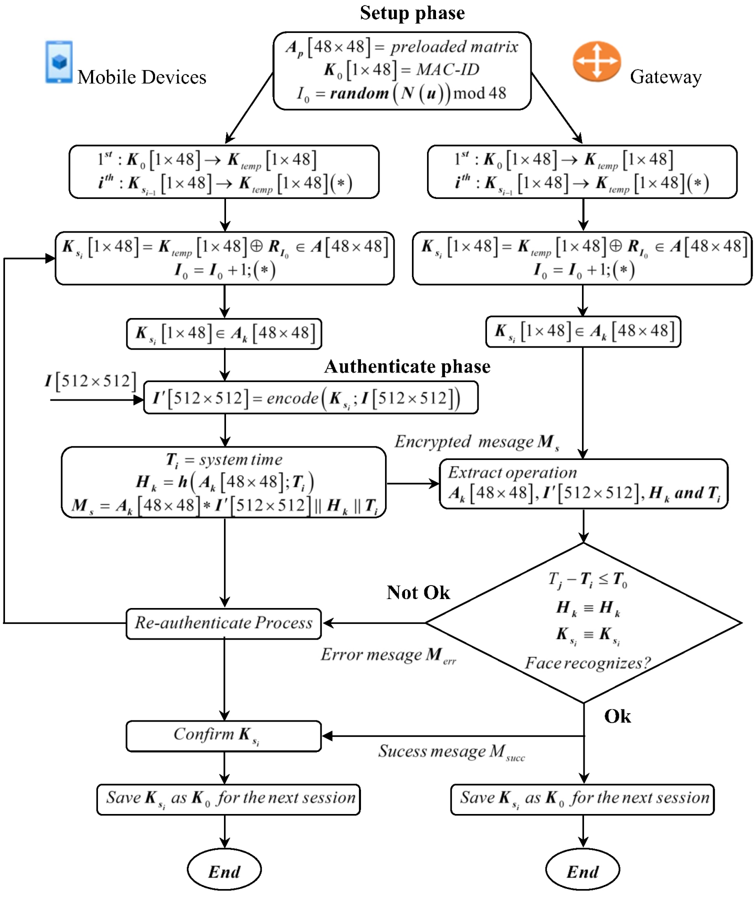

Setup phase
Mobile Devices
Gateway
$A_p[48\times 48] = preloaded\ matrix$
$K_0[1\times 48] = MAC\text{-}ID$
$I_0 = random(N(u)) \bmod 48$
$1^{st}: K_0[1\times 48] \to K_{temp}[1\times 48]$
$i^{th}: K_{s_{i-1}}[1\times 48] \to K_{temp}[1\times 48](*)$
$K_{s_i}[1\times 48] = K_{temp}[1\times 48] \oplus R_{I_0} \in A[48\times 48]$
$I_0 = I_0 + 1; (*)$
$K_{s_i}[1\times 48] \in A_k[48\times 48]$
Authenticate phase
$I[512\times 512]$
$I'[512\times 512] = encode(K_{s_i}; I[512\times 512])$
$T_i = system\ time$
$H_k = h(A_k[48\times 48]; T_i)$
$M_s = A_k[48\times 48] * I'[512\times 512] \| H_k \| T_i$
Encrypted mesage M_s
Extract operation
$A_k[48\times 48], I'[512\times 512], H_k\ and\ T_i$
$T_j - T_i \le T_0$
$H_k \equiv H_k$
$K_{s_i} \equiv K_{s_i}$
Face recognizes?
Not Ok
Re-authenticate Process
Error mesage M_{err}
Ok
Confirm K_{s_i}
Sucess mesage M_{succ}
Save K_{s_i} as K_0 for the next session
Save K_{s_i} as K_0 for the next session
End
End

Figure 2: Two-factor authentication diagram.

3.1.1 Setup Phase

In the setup phase, the gateway collected the Medium Access Control (MAC) address of the user's mobile and user's face images to store in its databases (key database and image database). The MAC address is a 48-bit length stored on the user's smartphone and the gateway to use as an initialization key K_0, represented by a vector $K_0[1 \times 48]$. The user and the gateway mutually store the following information:

- The initialization key $K_0[1 \times 48]$
- The random number representing the network size through the maximum number of devices, $N(u)$.
- A pre-agreed binary matrix $A_p[48\times 48]$ (preloaded matrix).
- A left bit shift algorithm is defined and used in the user device and the gateway.

The image database stored at the gateway is used to verify the match of the user's facial recognition parameters. This study uses the standard Support Vector Machine (SVM) algorithm for face recognition [6].

3.1.2 Authentication Phase

In this phase, several actions are required by both the mobile and gateway sides based on preset parameters in the previous phase. When a user wants to connect with the gateway, the user has to perform access actions as below.

- Taking the user's face image, encrypting the image, and embedding it to a message to send to the gateway:
  - The user's face image is captured by the smartphone's camera and transformed into an image matrix of size $I[512 \times 512]$. After that, the image matrix is encrypted by the session key, which is generated from the 2^{nd} authentication factor to preserve the confidentiality of the user image data, becoming the encrypted image, $I'[512 \times 512]$.
  - The mobile device creates the authenticated message (sending message) (M_s), which contains a session key, hash, and the encrypted image and sends it to the gateway at the given time.

- The gateway extracts the session key and user image from the received message M_s and executes the comparison processes between received parameters and predetermined parameters in the setup phase to recognize whether the authentication is successful. If the authentication is successful, the gateway sends the successful message (M_{succ}) to the mobile device. Otherwise, the gateway sends the error message (M_{err}) to the mobile device to retry the authenticated procedure.

Table 1: List of notations.

$A_p[48 \times 48]$	The preloaded random matrix
$A_k[48 \times 48]$	The embed key matrix
$I[512 \times 512]$	User's image matrix
$I'[512 \times 512]$	Coded User's image matrix
$K_0[1 \times 48]$	The Initial key as MAC address
$K_{temp}[1 \times 48]$	The temporal key
$K_{s_i}[1 \times 48]$	The session key
$K_{s_i}(g)[1 \times 48]$	The session key generated by the gateway
$N(u)$	The max. num. of mobile users connected to Gateway
h	Hash function
I_o	Initially Negotiated random number
I_u	User image captured by a camera
M_s	Encrypted message
M_r	Received message
M_{err}	Error message
M_{succ}	Successful message
R_{Io}	Round number I_o th in the A_p matrix

3.2 The Operation of the Authentication Scheme

The proposed authentication scheme performs the following algorithms during the authentication phase:

- **Algorithm 1:** The mobile user creates the session key and sends the encrypted message, M_s, to the Gateway.
- **Algorithm 2:** The gateway generates its session key and the embedded key matrix.
- **Algorithm 3:** The gateway verifies authentication parameters.
- **Algorithm 4:** The mobile device processes the notification messages from the gateway.

Table 1 lists the notations used in the paper.

3.2.1 The Mobile User Generates the Session Key, Creates and Sends the Message M_s to the Gateway

The mobile device performs the following computation steps described in Algorithm 1 to generate the session key. The calculation steps are described below:

- The mobile device uses the initialization key K_0, the matrix Ap and the random number $N(u)$ to generate the session key and create the message Ms to forward to the gateway.
- An I_o initialization number is generated by the same random number algorithm on both the mobile and gateway sides with the same value $N(u)$. We use $mod 48$, equivalent, $I_o \leq 48$ to ensure an efficient bit shifting algorithm.
- A random bit shift algorithm (Step 2) is applied to generate a temporary key, K_{temp}. The temporary key is XORed with the row whose index corresponds to I_o of the A_p matrix to generate a random session key, K_{s_i}.
- The session key Ks is embedded into the preloaded random matrix A_p to create the key matrix A_k (Step 3). Besides, the session key K_{s_i} is used as the symmetric key that encrypts the user's captured image to generate an encrypted image that is resistant to identity detection or forgery attacks (Step 4).
- The A_k matrix contains the session key embedded in the encrypted image matrix along with the integrity-preserving parameters H_k for the A_k matrix and the sending time parameter T_i to prevent modification or replay attack (Step 6, Step 7).
- The mobile device and the gateway agree on a timeout period against man-in-the-middle attacks, which is the time it takes for the mobile device to receive a response of the authentication status.

Algorithm 1 The mobile user generates the session key and sends the encrypted message (M_s) to the Gateway.

```
Input: the key $K_0$, matrix $A_p$, random number $i_o$, and the user's face image $I[512 \times 512]$ == User's image == Preloaded random matrix In the $1^{st}$ session: $K_0[1 \times 48]$ == MAC-ID; $i_0$ == random($N(u)$) mod 48. In the $i^{th}$ session: $K_0[1 \times 48] == K_{s_{i-1}}[1 \times 48]$; $i_0 == (i_0 + 1)$ mod 48
Output: the encrypted message $M_s = A_k[48 \times 48] * I'[512 \times 512] \| H_k \| T_i$
    Start
 1: Left shifting $i_0$ bits to create the temporal key
 2: Create the session key $K_{s_i}$ from $K_{temp}$ and matrix $A_p$: $K_{s_i}[1 \times 48] \leftarrow K_{temp}[1 \times 48] \oplus R_{I_o}[1 \times 48] \in A_p[48 \times 48]$
 3: Create the embedded key matrix $A_k$: $A_k[48 \times 48] \leftarrow K_{s_i}[1 \times 48] \oplus A_p[48 \times 48]$
 4: Use $K_{s_i}$ for encoding the matrix $I[48 \times 48]$ to matrix $I'[48 \times 48]$: $I'[512 \times 512] \leftarrow$ encode($K_{s_i}; I[512 \times 512]$)
 5: Create the time stamp $T_i \leftarrow T_i =$ current system time to determine the transmitted packet time.
 6: Create the hash value of the embedded key matrix $A_k$ to guarantee the integrity of the key matrix $A_k$ and the session key $K_{s_i}$: $H_k \leftarrow h(A_k[48 \times 48], \mathrm{T_i})$
 7: Create the encrypted message $M_s$ including $A_k$ and $H_k$ (from step 4 and 6): $M_s \leftarrow A_k[48 \times 48] \| * I'[512 \times 512] \| H_k \| T_i$
 8: Transmit $M_s$ to the gateway and wait for a $T_{timeout}$ for the acknowledgment: $T_t \leftarrow T_{timeout}$; $T_j - T_i \leftarrow T_t$
 9: if Acknowledgement $= M_{err}$ then
10:     Return to step 3
11: else{Acknowledgement $\leftarrow M_{succ}$}
12:     Go to the next step
13: end if
14: Store $K_{s_i}$ for the next authentication session
    End
```

Algorithm 2 The mobile user generates the session key and sends the encrypted message (M_s) to the Gateway.

```
Input: The key $K_0$, matrix $A_p$, random number $i_0$
    $A_p[48 \times 48] == $ preloaded matrix
    In the $1^{st}$ session: $K_0[1 \times 48] ==$ MAC-ID; $i_0 == \text{random}(N(u)) \bmod 48$.
    In the $i^{th}$ session: $K_0[1 \times 48] == K_{s_{i-1}}[1 \times 48]$; $i_0 == (i_0 + 1) \bmod 48$.
Output: The embedded key matrix containing $K_{si}(g)$
    Start
    Left shifting $i_0$ bits to create the temporal key $K_{temp}$
 1: Create the session key $K_{s_i}$ from the temporal key $K_{temp}$ and matrix $A_p$: $K_{s_i}[1 \times 48] \leftarrow K_{temp}[1 \times 48] \oplus R_{I_o}[1 \times 48] \in A_p[48 \times 48]$
 2: Create the embedded key matrix $A_k$: $A_k[48 \times 48] \leftarrow K_{s_i}[1 \times 48] \oplus A_p[48 \times 48]$
 3: Store $A_k$ and $K_{s_i}$
    End
```

3.2.2 The Gateway Generates its Session Key and the Embedded Key Matrix

The gateway maintains two databases (key database and image database) for performing the authentication:

- Database of user's face images: when a user registers to use the service with the gateway, the user takes some face images, and the gateway stores the user's face images as biometric information.
- The preloaded matrix A_p and the MAC address of each mobile device are known by both a mobile user and the gateway.

The session key generation in the gateway is carried out using Algorithm 2, as described below.

3.2.3 The Gateway Verifies Authentication Parameters

After receiving the message M_s from the user's mobile device, the gateway performs the watermark decoding procedure to extract the user-encoded image matrix, $I'[512 \times 512]$, the embedded key matrix A_k, the hash function H_k and the message delivery time T_i. Note, H_k and T_i sent as plain text.

The gateway compares the message receipt time to determine if the message is valid or expired in step 3 of Algorithm 2.

The gateway uses the session key $K_{s_i}(g)$ obtained after performing the Algorithm 2 to match the key K_{s_i} in the matrix A_k sent from the mobile device. If there is a match, the key K_{s_i} is used to restore the captured user image, I_u.

The gateway authenticates the user image using the SVM algorithm to confirm face recognition parameters. The gateway then sends either M_{err} or M_{succ} to the mobile user in the case of authentication failure or success, respectively.

Algorithm 3 The gateway performs the verification of authentication parameters.

```
Input: Encrypted message $M_s$
Output: Verification of the authentication scheme
    Start
 1: Receive the image message $M_s$: $M_s = A_k[48 \times 48] || * I'[512 \times 512] || H_k || T_i$
 2: Detach components of $M_s$ to $A_k[48 \times 48]$, $I'[512 \times 512]$, $H_k$, and $T_i$
 3: Verify timeout value over received time $T_j$
 4: if $T_j - T_i > T_0$ then
 5:     Send the error message $M_{\text{err}}$ to the mobile user
 6: else
 7:     go to next step
 8: end if
 9: Check the hash $H_k$
10: if $H_k \neq H_k(g)$ then
11:     Send the error message $M_{\text{err}}$ to the mobile user
12: else
13:     go to next step
14: end if
15: Authenticate the session key $K_{\text{si}}$
16: if $K_{\text{si}} \neq K_{\text{si}}(g)$ then
17:     Send the error message $M_{\text{err}}$ to the mobile user.
18: else
19:     go to next step
20: end if
21: Use the session key to recover the captured user image $I[512 \times 512] = \text{decode}(K_{\text{si}}, I'[512 \times 512])$
22: Use the SVM algorithm to authenticate the user's face
23: if user's face is not recognized then
24:     Turn the error message $M_{\text{err}}$ to the mobile user.
25: else{Send the successful message $M_{\text{succ}}$ to the mobile user}
26:     $M_{\text{succ}}(s) = \text{encode}(M_{\text{succ}}, K_{\text{si}})$
27: end if
    end
```

3.2.4 The Mobile User Processes the Response Message Received from the Gateway

After the mobile user receives the authentication error message M_{err} from the gateway, it performs Algorithm 1 again for re-authentication. In the case of receiving M_{succ}, the mobile device knows that the session key is secured, then it can be used as the initialization key for the following authentication session. Algorithm 4 below describes how the mobile user processes the response message received from the gateway.

4 Security Analysis

In this section, we evaluate the security strength of the proposed authentication scheme by doing the following analysis:

- Security protection against widespread security attacks.
- Security evaluation using the BAN logic model.

Algorithm 4 The mobile user processes the response message received from the gateway.

```
Input: Received messages ($M_{err}$, $M_s ucc$)
Output: The authentication is confirmed
   Start
1: Receive the returned acknowledgment message
2: if Acknowledgement = $M_{err}$ then
3:     Perform Algorithm 1 again for re-authentication
4: else[$Msucc = decode(K_{s_i}, M_{succ}(s))$]
5:     go to step 7
6: end if
7: Confirm $K_{s_i}$ is safe
   end
```

- Security evaluation using the simulation tool AVISPA.

4.1 Security Protection Against Security Attacks

The proposed authentication scheme can protect the communications between mobile users and the gateway against the common security attacks below. *a. Security attacks on mobile users*

- Impersonation Attack: Impersonation attacks aim to fake device parameters. However, such an attack in this scheme is not possible when the session key K_{s_i} is changed dynamically on every session. the probability of finding the key K_{s_i} in the matrix A_k is $1/2^{48}$.
- User Credentials Attack: because the proposed scheme provides two-factor authentication simultaneously, it will reduce the possibility of attacks on user images. Furthermore, the user's identity will not be revealed because the user's image is encrypted during communication from the mobile device to the gateway.
- Attack on session keys: In this proposed scheme, the session key is secretly protected by matrix A_p and hidden in the captured user image. Furthermore, the session key is integrity protected by the one-way Hash function and the session time limit. So the secrecy of the session key (K_{s_i}) is guaranteed.

b. Security attacks on communications links

- Replay attack: The attack repeats through spoofing packets transmitted from the mobile device to the gateway. However, the entire outgoing message is encrypted, and each packet has a unique identifier that comes from the hash value of the encryption matrix (A_k) and the sending message time T_i. So, the repeat attack is defeated by this item.
- Eavesdropping attack: Eavesdropping attacks illegally collect information from packets transmitted by mobile devices over the air. The proposed scheme changes the session key, followed by its previous safe state. Hence, it ensures the security of the session key (K_{s_i}), and the authentication information is securely encrypted against eavesdropping attacks.
- Man-in-the-Middle attack: The proposed scheme is protected by the device identifier, the session key, and the hash. So, this attack is defeated.

4.2 BAN Logic Analysis

BAN logic was developed by Burrows, Abadi, and Needham [5], which includes a set of rules to design, develop, and validate security schemes. We have applied the BAN logic to test the correctness of the security protocol and determine the trustfulness of agreement among the participants in the proposed authentication scheme. The following notations are used in the BAN logic:

- $A| \equiv X$: A believes the statement X.
- $A \triangleleft X$: A sees X, i.e. A has received a message containing X.
- $A| \sim X$: A once said X i.e $A \equiv X$ when A sent it.
- $A| \Rightarrow X$: A has authority or jurisdiction over X.
- $\#(X)$: X is a fresh message.
- $A \leftrightarrow B$: K is the shared secret key between A and B.
- X_K: X is encrypted with key K.
- $< X >_Y$: formula X is combined with formula Y.
- $(X)_K$: X is hashed with key K.
- (X, Y): X or Y is one part of formula (X; Y).

The logical postulates in the BAN logic are described using the below-mentioned rules:

- Rule 1 (Message Meaning Rule (MMR)): P believes Q once said X if P sees a message X encrypted with K, and P believes K is a shared secret between P and Q.

$$\frac{P| \equiv P \overset{K}{\longleftrightarrow} Q, P \triangleleft \{X\}_K}{P| \equiv Q| \sim X}$$

$$\frac{P| \equiv P \overset{Y}{\longleftrightarrow} Q, P \triangleleft \langle X \rangle_Y}{P| \equiv Q| \sim X}$$

Rule 1 satisfies the proposed scheme because the key K_{s_i} is secretly shared between the mobile device and the gateway via the A_p matrix and the bit shifting method. When the gateway receives the encrypted M_s, it believes that K_{s_i} is a good and secret key associated with the identity generated from the mobile device's MAC ID. Here, P is the gateway, and Q is the mobile device.

- Rule 2 (Nonce Verification Rule (NVR)): P believes Q believes X if P believes Q once said X and P believes X is fresh

$$\frac{P| \equiv \#\{X\}, P| \equiv Q| \sim X}{P| \equiv Q| \equiv X}$$

Rule 2 is satisfied because the current belief of the i^{th} session key is confirmed by the previous $(i-1)^{th}$ session having been successfully authenticated.

- Rule 3 (Jurisdiction Rule (JR)): P believes X if P believes that Q believes X and P believes Q has jurisdiction over X.

$$\frac{P| \equiv Q| \equiv X, P| \equiv Q| \Rightarrow X}{P| \equiv X}$$

  Assuming the index I_o is established in the initial secure phase, the gateway can fully infer to believe the session key generated by the mobile device.

- Rule 4 (Freshness Rule (FR)): The entire formula is believed to be fresh if a part of the formula is believed to be fresh.

$$\frac{P| \equiv \#\{X\}}{P| \equiv \#\{X, Y\}}$$

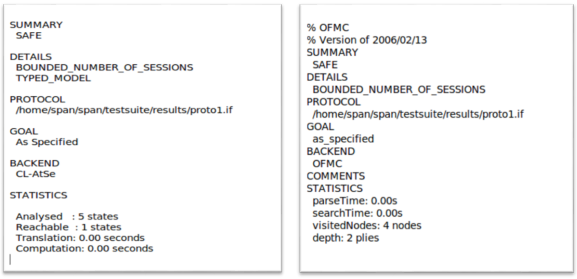
P believes combined formula (X; Y) if P believes X and P also believe Y.

$$\frac{P| \equiv X, P| \equiv Y}{P| \equiv (X, Y)}$$

Authentication sessions are sequenced concerning each other in order. The successful response message is the basis for generating the next session key, so rule 4 of BAN is satisfied in this proposed scheme.

4.3 Security Assessment using AVISPA Tool

AVISPA is a powerful tool for the Automated Validation of Internet Security Protocols and Applications [16]. The tool uses modules and an expressive formal language to detail protocols and security properties in state-of-the-art automatic analysis techniques. AVISPA comprises state-of-the-art backend models such as CL-AtSe and OFMC [BACKEND]. These backends perform various automatic analyses to detect vulnerabilities in the security scheme. It uses the formal language High-Level Protocol Specification Language (HLPSL) to code a specified security algorithm, and a translator known as HLPSL2IF is used to convert the HLPSL code into the Intermediate Form (IF) and then bring out the results. We have used HLPSL to test Algorithms 1 and 2 and get results, as shown in Figure 3. Figure 3 shows the output format generated by AVISPA's OFMC and CL-AtSe backends. SUMMARY generally shows whether a security scheme being tested is safe or unsafe. In our case, it presents as a safe condition.



```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/proto1.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 5 states
  Reachable  : 1 states
  Translation: 0.00 seconds
  Computation: 0.00 seconds
```

Results of back-end CL-AtSe

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/proto1.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.00s
  visitedNodes: 4 nodes
  depth: 2 plies
```

Results of back-end OFMC

Figure 3: Analysis results using AVISPA.

5 Authentication Testbed

Based on the above-proposed algorithms, we implemented the testbed model, as shown in Figure 4, for a study case of the smart door lock application. The mobile devices in the testbed were Samsung A50, Samsung Note10, and Oppo Reno, which use the Android 9.0 operating system. The gateway is implemented using Raspberry 3 with OS version 4.19. We use the Mosquitto library to install the MQTT Broker service to connect the gateway and a mobile phone over the IEEE 802.11g link. Authentication messages are transferred by MQTT protocol.

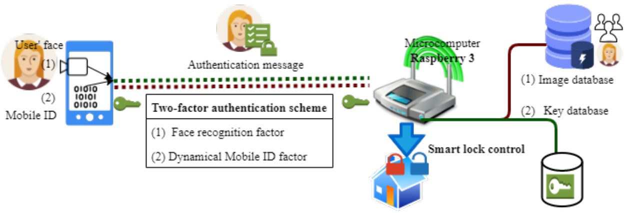


Figure 4: Authentication testbed model.

The authentication software performs the session key generation on the user device side, captures the user image via the camera, encrypts the image, and sends the authentication message (M_s) to the gateway. On the gateway side, the gateway recognizes the user's face by applying the SVM algorithm on the image database at $ageitgey/face-recognition$ [1] and uses the algorithm proposed above to authenticate the session key. We have made 100 trials to collect run-time values of the session key, facial image encryption, and facial image decryption processes with various tested phones. Figure 5 shows the higher time value for generating the session key is 1.4 seconds with the Reno phone, and the lower time is 0.6 seconds with the Note 10 phone.

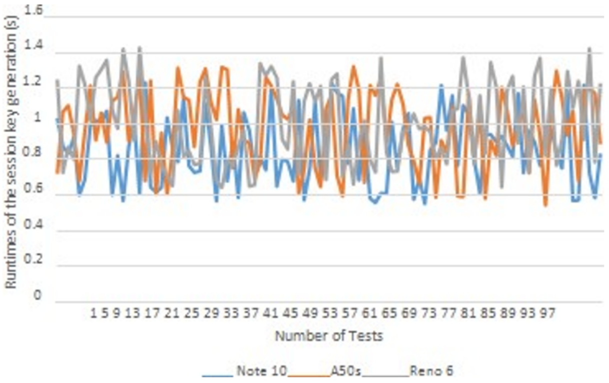


Figure 5: Run times of session key generation on the tested phones.

Table 2 summarizes the features of our proposed scheme compared with the Multi-factor authentication

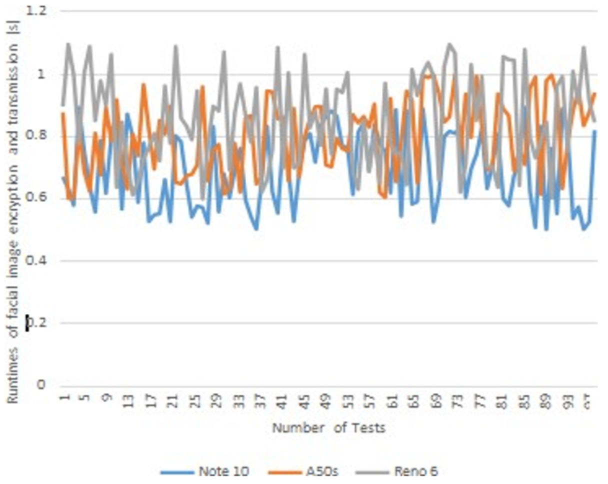


Figure 6: Run times of facial image encryption processes.

Table 2: The comparison features.

No.	Security feature	[9]	[10]	[11]	[12]	[13]	Ours
1	User impersonation attack	Yes	Yes	No	Yes	Yes	Yes
2	User untraceable	Yes	Yes	Yes	Yes	Yes	Yes
3	Sensor node impersonation attack	Yes	Yes	Yes	Yes	Yes	Yes
4	Mutual authentication	Yes	Yes	No	Yes	Yes	Yes
5	Secret key agreement	Yes	Yes	Yes	Yes	Yes	Yes
6	Replay attack	Yes	No	No	Yes	Yes	Yes
7	User device stolen attack	Yes	No	No	No	Yes	Yes
8	Denial of service attack	Yes	Yes	Yes	Yes	Yes	Yes
9	Password guessing attack	Yes	No	No	Yes	Yes	Yes
10	User anonymity	Yes	No	Yes	Yes	Yes	Yes
11	Sensor node anonymity	Yes	Yes	Yes	Yes	Yes	Yes
12	Forward Secrecy	Yes	Yes	Yes	Yes	Yes	Yes
13	Session key attack	Yes	No	Yes	Yes	Yes	Yes
14	Gateway node bypass attack	Yes	No	Yes	Yes	Yes	Yes
15	Insider attack	Yes	No	Yes	No	Yes	Yes
16	Different random session key	No	No	No	No	No	Yes
17	BAN logic	N/A	N/A	N/A	N/A	N/A	Yes
18	AVISPA	N/A	N/A	N/A	N/A	N/A	Yes

schemes of other authors. Besides the apparent advantages of friendliness, ease of use, and lightness, the scheme provides a higher level of security thanks to the random session key generation.

On the gateway side, the run times on the Raspberry 3 for decoding encrypted message M_s is approximately 1.1 seconds, as shown in Figure 6. These test results show that the average time of the whole authentication process is approximately 3.0 seconds. The test results prove that the proposed authentication scheme can effectively provide smart door lock services for smart homes in real applications.

6 Conclusion

This paper proposes a two-factor authentication scheme for communication and smart control systems. The proposed scheme provides a user-friendly, secure, lightweight approach through face recognition, dynamic session key generation, and watermarking techniques. Furthermore, the random dynamic key embedded in the captured user image reduces the key distribution procedure and ensures the privacy of the user image. Our proposed scheme has been analyzed to illustrate its security strength under the standard attacks and ensure that BAN logic rules are met. Furthermore, the solution is also verified through the VISPA tool to prove the proposal's safety. Besides, to determine the solution's effectiveness in the actual application setting, the proposed scheme is deployed for a smart door lock application, a typical application in smart home systems. The experimental results show the authentication execution time is acceptable for the application, which does not strictly require real-time. In our near future work, several advanced AI-based recognition solutions will be integrated into the scheme to reduce the authentication time and extend to new real-time scenarios.

References

[1] Face_recognition. https://github.com/ageitgey/face_recognition. Accessed: 12 Jan 2022.

[2] Abdulla, A. I., Abdulraheem, A. S., Salih, A. A., Sadeeq, M., Ahmed, A. J., Ferzor, B. M., Sardar, O. S., and Mohammed, S. I. Internet of things and smart home security. *Technol. Rep. Kansai Univ 62*, 5 (2020), 2465–2476.

[3] Aparna, J., and Ayyappan, S. Image watermarking using diffie hellman key exchange algorithm. *Procedia Computer Science 46* (2015), 1684–1691.

[4] Bal, S. N., Nayak, M. R., and Sarkar, S. K. On the implementation of a secured watermarking mechanism based on cryptography and bit pairs matching. *Journal of King Saud University-Computer and Information Sciences 33*, 5 (2021), 552–561.

[5] Burrows, M., Abadi, M., and Needham, R. M. A logic of authentication. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences 426*, 1871 (1989), 233–271.

[6] Cortes, C., and Vapnik, V. Support-vector networks. *Machine learning 20*, 3 (1995), 273–297.

[7] Halim, M. A. A., Othman, M. F. I., Abidin, A. Z. Z., Hamid, E., Harum, N., and Shah, W. M. Face recognition-based door locking system with two-factor authentication using opencv. In *2021 Sixth International Conference on Informatics and Computing (ICIC)* (2021), IEEE, pp. 1–7.

[8] Hoang, T.-M., Bui, V.-H., and Nguyen, N.-T. A lightweight multi-factor authentication scheme based on digital watermarking technique. In *2021 International Conference on Advanced Technologies for Communications (ATC)* (2021), pp. 270–274.

[9] Kabir, M., et al. An efficient low bit rate image watermarking and tamper detection for image authentication. *SN Applied Sciences 3*, 4 (2021), 1–17.

[10] Kaur, D., and Kumar, D. Cryptanalysis and improvement of a two-factor user authentication

scheme for smart home. *Journal of Information Security and Applications 58* (2021), 102787.

[11] Mohanarathinam, A., Kamalraj, S., Prasanna Venkatesan, G., Ravi, R. V., and Manikandababu, C. Digital watermarking techniques for image security: a review. *Journal of Ambient Intelligence and Humanized Computing 11*, 8 (2020), 3221–3229.

[12] Munir, A., Ehsan, S. K., Raza, S. M., and Mudassir, M. Face and speech recognition based smart home. In *2019 International Conference on Engineering and Emerging Technologies (ICEET)* (2019), IEEE, pp. 1–5.

[13] Panwar, N., Sharma, S., Mehrotra, S., Krzywiecki, Ł., and Venkatasubramanian, N. Smart home survey on security and privacy. *arXiv preprint arXiv:1904.05476* (2019).

[14] Shuai, M., Yu, N., Wang, H., and Xiong, L. Anonymous authentication scheme for smart home environment with provable security. *Computers & Security 86* (2019), 132–146.

[15] Sureshkumar, V., Amin, R., Vijaykumar, V., and Sekar, S. R. Robust secure communication protocol for smart healthcare system with fpga implementation. *Future Generation Computer Systems 100* (2019), 938–951.

[16] Vigano, L. Automated security protocol analysis with the avispa tool. *Electronic Notes in Theoretical Computer Science 155* (2006), 61–86.

[17] Wahyudono, B., and Ogi, D. Implementation of two factor authentication based on rfid and face recognition using lbp algorithm on access control system. In *2020 International Conference on ICT for Smart Society (ICISS)* (2020), IEEE, pp. 1–6.

[18] Wazid, M., Das, A. K., Odelu, V., Kumar, N., and Susilo, W. Secure remote user authenticated key establishment protocol for smart home environment. *IEEE Transactions on Dependable and Secure Computing 17*, 2 (2017), 391–406.

[19] Wu, F., Xu, L., Kumari, S., and Li, X. An improved and anonymous two-factor authentication protocol for health-care applications with wireless medical sensor networks. *Multimedia Systems 23*, 2 (2017), 195–205.

[20] Xiong, W., Zhou, F., Wang, R., Lan, R., Sun, X., and Luo, X. An efficient and secure two-factor password authentication scheme with card reader (terminal) verification. *IEEE Access 6* (2018), 70707–70719.

[21] Yuan, B., Du, C., Wang, Z., and Zhu, R. Research on intelligent algorithm of identity authentication based on facial features. *Wireless Communications and Mobile Computing 2021* (2021).

[22] Zhang, J., Tan, X., Wang, X., Yan, A., and Qin, Z. T2fa: Transparent two-factor authentication. *IEEE Access 6* (2018), 32677–32686.