Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2022.021297
Article

Tech Science Press

GDPR Compliance IoT Authentication Model for Smart Home Environment

Hisham Raad Jafer Merzeh[1,*], Mustafa Kara[2], Muhammed Ali Aydın[3] and Hasan Hüseyin Balık[1]

[1]Yildiz Technical University Graduate School of Science and Engineering Faculty of Electrical and Electronics Engineering,
Istanbul, 34220, Turkey
[2]National Defense University, Hezarfen ASTIN Computer Engineering Department, Istanbul, 34000, Turkey
[3]Istanbul University-Cerrahpaşa Faculty of Engineering, Istanbul, 34320, Turkey
*Corresponding Author: Hisham Raad Jafer Merzeh. Email: hisham.raad.jafer.al-assedy@std.yildiz.edu.tr

Received: 28 June 2021; Accepted: 29 July 2021

Abstract: The Internet of things (IoT) became quickly one of the most popular and most discussed topics in research. Studies paid attention to the Internet stuff, primarily to new products that aim to achieve greater efficiency and simplicity in life. IoT may cover several fields of the smart environment. Because of the data exposure that occurs when data is transferred via various channels, data protection issues have become a major problem as the company continues to expand. When user privacy and property are taken into consideration, the situation may become much worse. As a result, the authentication process for communicating entities has garnered considerable attention. In this paper, we proposed a secure authentication model for smart home applications, which privacy considered and complies with the General Data Protection Regulation GDPR. The proposed scheme improved the existing authentication schemes' performance and security level. This work based on the Elliptic curve cryptography ECC, one-way hash function, and XOR operation. The proposed lightweight authentication model is suitable for resource-constrained devices. This study is developing the offline direct authentication model to authenticate users and IoT devices in the local network. In addition, our scheme uses the online authentication server to authenticate all system parts.

Keywords: IoT; internet of things; authentication; smart home

1 Introduction

The growing number of connected devices form what is called the Internet of Things. It is an extensive network of networks that connects smart devices such as sensors and actuators. These devices are supported in various areas such as smart cities, public health, smart houses, intelligent transport, energy management, smart grids, agriculture, and waste management, etc. The constraints and requirements of devices that are connected raise a wide range of challenges, including connection challenges for the massive number of smart devices to exchange data with each other and security challenges with the need to protect the Internet of things networks from being attacked. At the same time, the need for protection from exploitation to be used as an attacking tool. These challenges are increased with the nature of resources-limited Internet of Things devices. That makes traditional communication protocols and security systems ineffective and even not possible for the Internet of Things. Security issues related to IoT have become more worrying because IoT devices are everywhere and involved in critical applications. Worsening and aggravating the impact of any security breach to the extent of being life-threatening [1]. The nature of the applications that IoT network service specify the security requirements. The need for confidentiality, integrity, and authentication depends on the application requirements of the security level. In particular, authentication is a fundamental requirement for IoT. Trusting the devices connected in an IoT network is an essential principle for a trusted and secure network [2]. Only one infected node can become malicious and bring down the whole system or cause a catastrophic effect [1]. To minimize these problems, the data must be processed using encryption and decryption operation based on secure key-based algorithms to send them securely from one node to another. These algorithms can reduce the encryption and decryption time effectively and efficiently, make the cracking process more difficult and time-consuming [3]. This study proposed an authentication schema for the user to access the system using a web browser or mobile application, also the server authentication between the central server and local (edge) server, and our scheme has M2M authentication between the local server and IoT devices. All these will use an online authentication server. In the case of a local network connection, the user can access the smart devices through the local server. Our scheme shows an offline three-way authentication model for authenticating the user using a smart card and QR code for registration and a secure token for mutual authentication.

We summarize the contribution as follows:

- This study focused on authentication protocols for IoT devices in the smart home environment.
- This study proposed an authentication system to secure access to smart home devices. It complies with GDPR that concern with user privacy. In addition, support for offline authentication for devices to be used in the local networks.
- This study proposed a lightweight model with high performance and low computation cost, which suitable for resource-constrained devices. The emphasis on lightweight security is essential because IoT devices are generally limited resources, with restricted memory, communication and computation capabilities, and battery power.
- Finally, the proposed protocol has many security features and resistance to different types of security attacks.

The rest of the paper is organized as follows, the related work is discussed in Section 2, system module and proposed model discussed in Section 3, security analysis and formal verification in Section 4, the performance evaluation in Section 5, and the conclusion in Section 6.

1.1 General Data Protection Regulation (GDPR)

GDPR stands for General Data Protection Regulation. The General Data Protection Regulation (EU) 2016/679 (the "GDPR"). The GDPR adopted by the European Parliament on April 14, 2016, and took effect on May 25, 2018. This regulation repealed the Data Protection Directive 95/46/EC (the "Directive") that applied in 1998, which is concerned with the protection of people's personal data processing and mobility freedom of that data [4]. It improved individuals' data privacy and data rights while also expanding data controllers' roles and duties. Personal data is all information about a recognized or identifiable real person. A real person is said to be identifiable if he can be either directly or indirectly identified [5]. The GDPR applies to every enterprise that handles the personal data of EU individuals or analyzes their activity within the EU wherever they are [6]. What are the different types of personal data? The GDPR outlines certain kinds of personal data (sensitive data) that must be secured with additional safeguards and should not be collected without specific authorization, justification, or a few other exceptions. Those categories are Racial or Ethnic Origin, which refers to the cultural origins of the person's ancestors. Political opinions of an individual on a specific topic or voting intention. Religious or philosophical beliefs of individuals and their related behaviors. A trade union or any social membership. Genetic data related to a person. Biometric data of an individual. People's health records. Personal sex life behaviors. An individual's sexual orientation [7,8]. See Tab. 1.

Table 1: Different type of personal data

Nonpersonal data	Personal data	Sensitive personal data
Address without a name	Name and address	Racial or ethnic origin
A generic email address such as info@web.com	Personal email address Personal telephone number	Political opinions
A receipt with date, time, last 4 digits of credit card number but no name or email address	Name and last 4 digits of the credit card number	Religious beliefs
Corporate accounts with summary payroll data	Pay records with gender and age even if without a name	Sexual preferences
Company name and website	A web cookie	Biometric information

1.2 Taxonomy of Authentication Schemes

The IoT authentication schemes taxonomy is given in this section. This taxonomy is implemented using different selection principles according to the main features and the characteristic matches of authentication schemes. The implementation of the authentication scheme in different layers will change the authentication techniques accordingly. Fig. 1. summarizes the criteria by describing and outlining IoT authentication schemes as follows [9].

The authentication factors: the Identity Information is submitted to a party by another to authenticate oneself. Identity may consist of one or more data produced with a Hash function, random number generation functions, bitwise operations, and cryptography algorithms to be utilized by identity-based authentication schemes. The Context could be physical, based on the physical characteristics of an individual to represent the biometric information, for example, fingerprints scanning data, hand geometry data, retinal scans of individual, etc. Or context could be behavioral, based on an individual's behavioral characteristics to represent Biometric data, e.g., keystroke dynamics (rhythm pattern and timing of a person when he is typing or using an input device), gait analysis (the technique used to evaluate how we walk or run), voice ID (voice authentication using voiceprint of an individual), etc.

Use of tokens: it is either a Token-based authentication Identification token (a piece of data) is created by a server such as OAuth2 protocol or open ID is the key to authenticating a user/device. Or Non-token-based authentication means that The user has to use the username and password for authentication in every data exchange session.

The authentication procedures: there is One-way authentication means that The first party authenticates the second party in order to communicate with each other. At the same time, the first party remains unauthenticated. Also, we have Two-way authentication In this case, both entities authenticate each other, it is also called mutual authentication. Or Three-way authentication this authentication is based on a third party as a central authority. It helps two communicated entities to authenticate themselves mutually.

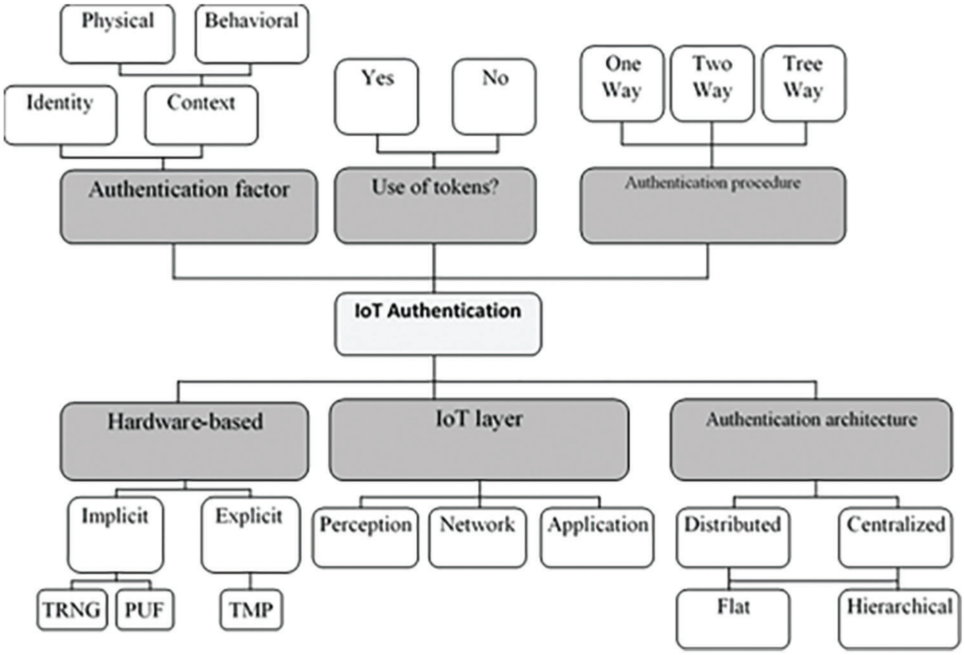


Figure 1: Taxonomy of IoT authentication schemes [9].

The authentication architectures: there is Distributed which means using a method of distributed direct authentication among the communicating parties. Or centralized in order to distribute and manage the credentials that are used for authentication, a centralized server or a trusted third party is used. The authentication scheme's architecture, whether centralized or distributed, can be Hierarchical using a multi-level architecture to handle the authentication process, or Flat using no hierarchical architecture in order to deal with the authentication procedures.

The IoT layers: Specifies the layer that is applied to the authentication procedure. In the Perception layer, the end nodes in the IoT platform use this layer to acquire, manipulate, and digitize data that they perceive. The perception layer send perceived data to the Network Layer which is responsible for acquiring and processing them. The application layer is responsible for data reception from the network layer and the provision of requested user services.

Hardware-based: The process of authentication can involve the use of physical characteristics of the hardware chipset or using the hardware itself. Like The Implicit hardware-based to improve the authentication, the physical features of the hardware chipset are used like Physical Unclonable Function (PUF) or True Random Number Generator (TRNG). Or the explicit hardware-based Trusted Platform Module (TPM) is a hardware chip dedicated to storing and processing hardware authentication keys. It is an essential base in some authentication systems [9].

1.3 Authentication Protocol Phases

Many phases must be involved in the authentication schemes some of these phases are essential and some of them can be replaced with another phase as shown in Fig. 2. In the beginning, there is a setup phase or sometimes called the installation or precomputation phase. This phase is used to generate secrete keys, IDs, initial values of some parameters, and a pre-shared key can be installed in this phase also. The registration phase is used to register the user or the device in the server database and store the identification data for the first time. To be used for authentication. The login phase is the phase that the user or the device request to access the system and the server receive this request to start the authentication phase.

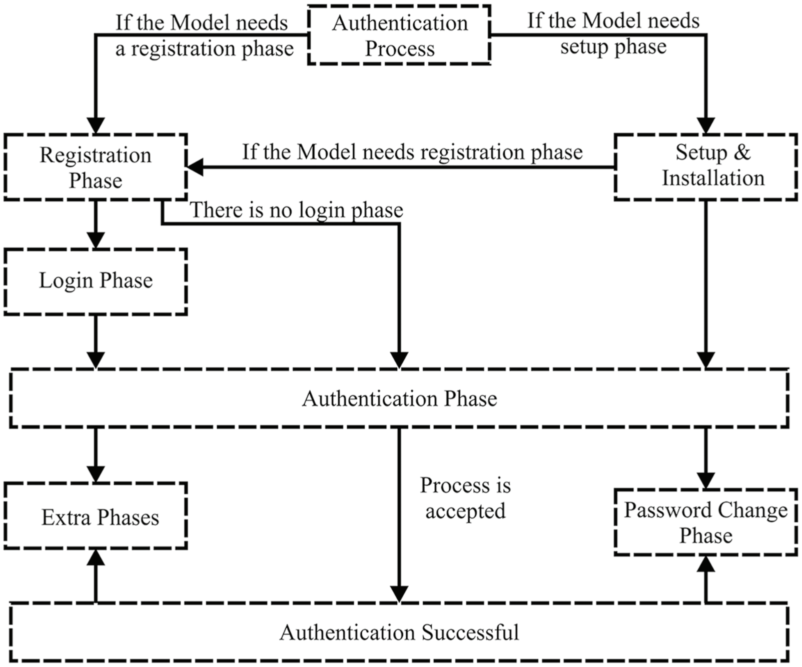


Figure 2: Authentication protocol phases

The authentication phase is the phase that the user provides the authentication factor to be checked by the server and authenticate. Some schemes are combined these two phases into one phase. After the authentication phase is approved. The password change phase can be used for password changing according to the expiration time of the password that is proposed in some schemes. Also, there are some extra phases is needed in many schemes to complete the functionality of the authentication system. For example, Revocation and re-registration phased, logout phase, home Gateway rejoin network phase, and some other different extra phases.

2 Related Work

The widespread of smart devices and the Internet of things in various areas of daily life has led to the emergence of various new security problems, especially privacy concerns. Unfortunately, most of the systems responsible for managing these devices may lack appropriate protection to protect sensitive information, and this may be due to resource, cost, or design constraints. Therefore, we need systems that can manage and protect the user, smart IoT devices from any attempt to breach security, disclose sensitive information, and violate the privacy of the user. One of the most important aspects that must be taken into account and attention is to ensure the identity of the user and make sure of the devices connected to the system. Therefore, there are many studies and research published aimed at solving these security problems or reducing their risk. Kalra et al. [10], Chang et al. [11], Wang et al. [12], and Wu et al. [13] based on ECC Elliptic Curve Cryptography to design a secure authentication protocol and securing the communication between an embedded device and cloud server. They are based on a two-way authentication procedure in the IoT application layer, centralized and flat authentication architecture is used, and token-based protocol. the setup phase is considered in these methods except in models of

Wang et al. [12] and Wu et al. [13], while other phases like registration, login, and authentication are used. Password change and any additional phased are not considered in these proposed schemes. Unfortunately, there are some weaknesses in Kalra et al.'s scheme [10], like the failure of mutual authentication, the session key ambiguity, and guessing the password attack. They are immune to some types of attacks like MITM, Reply attacks, but some of the attacks like timing attacks, forgery attacks, and DoS are not considered. Shuai et al. [14], Yu et al. [15], Naoui et al. [16], and Hussain et al. [17] proposed a remote authentication scheme. They are using Identity as an authentication factor with a three-way authentication procedure; the application layer is responsible for authentication in these schemes. They are using a Flat and centralized architecture authentication system. Different types of authentication phases are used such as installation, registration, login, and authentication, and password change phases except in Yu et al. [15] scheme's the password change is not used. the schemes of Shuai et al. [14] and Naoui et al. [16], needs a secure channel for registration, and they are based on ECC encryption, while in the schemes of Yu et al. [15] and Hussain et al. [17], they are used pre-shared key for encryption and registration, also they are used pre-shared key for encryption and registration, also they based on XOR operation and one-way hash function. These schemes are immune to some types of attacks However, they are not immune to the DoS attack. Some of these methods have higher computational and communication costs. Lyu et al. [18], Coruh et al. [19], and Sun et al. [20] proposed secure authentication and communication scheme based on ECC encryption for smart home and machine-to-machine systems. ECDH key exchanged is used in Coruh et al. [19] and Sun et al. [20]. The secure channel is required for registration in Lyu et al. [18] and Sun et al. [20]. These schemes are immune to many types of security attacks. However, there are some drawbacks in Lyu et al. [18] and Sun et al. [20] like the DoS attack. Kumar et al. [21] proposed an authentication scheme base on IPv6 headers for unique addressing and identification, in addition to the Bio ID, Smart card, and Passwords. Based on symmetric encryption, Bio hashing and, a one-way hash function for authentication. In all previously listed works, we found that there is no mention of the authentication system model in the offline environment for a smart home. This means that you cannot use these models without internet service existence. There are no alternative ways for authentication and secure communication system, which can be used from inside the smart home without the need for internet service. We proposed in this paper some solutions that can be used for authentication and secure communication in the offline environment using a local network only. We compared our work with the recent similar works and proposed a new model that improved their drawbacks. The study discuss the main features of each similar work showing their advantages and disadvantages. Wang et al.'s scheme (in 2017) [12], their model includes the registration phase and Authentication phase. Used real device ID for registration and Authentication. Need a secure channel for registration. Base on ECC key pairs, scalar multiplication of the ECC point, hash function, XoR operation. Computational complexity costs 209.0221 ms. Using 3 messages for authentication with a Total size of 1568 bit. Can resist some security threats but insecure for Denial of Service (DoS) attack, Eavesdropping Attacks, and Forward Secrecy attack. No consideration for Data Secrecy and Reliability, Anonymity, Privacy, and GDPR Compliance. In Wu et al.'s scheme (in 2020) [13], Their model includes the registration phase and authentication phase. Used real device ID for registration, pseudo ID for authentication. Need a secure channel for registration. Base on ECC key pairs, modular exponential of the ECC point, bilinear mapping, scalar multiplication of the ECC point, hash function, XoR operation. Computational complexity costs 164.5702 ms. Using 3 messages for authentication with a Total size of 1404 bit. Can resist some security threats but insecure for Denial of Service (DoS) attack, and forward secrecy attack. No consideration for data secrecy and reliability, anonymity, privacy, and GDPR compliance. Kumar et al.'s scheme (in 2020) [21], their model includes initial, addressing, registration, authentication, session agreement, and password update phase. Based on IPv6 addressing schemes. Use a smart card, password, and user biometric identity for Authentication. Use biometrical hashing, symmetric encryption/decryption, and hash function. Computational complexity costs 209.0221 ms. Using 5 messages for authentication with a

Total size of 959 bit. Can resist some security threats. No consideration for privacy, and GDPR compliance. According to these features of similar works, our scheme shows higher performance, low computation cost, and resistance to more security threats. The comparison and performance evaluation is explained in detail in Section 6.

3 System Module

To guarantee that smart devices in the IoT systems can be trusted to be what they claim to be, strong IoT device authentication is needed. As a result, each IoT device needs a distinct identity that can be verified when it tries to connect to a gateway or central server. IT system managers can follow each device throughout its existence, communicate securely with it, and prohibit it from doing dangerous operations with this unique ID in place. Administrators can easily withdraw a device's privileges if it exhibits unusual behavior.

3.1 Assumption

Each smart home device has a smart card contain device universal unique ID with a public key sored in the card for registration. This card is issued in manufacture time. Every user device also needs a smart card contains a universally unique ID, this ID is generated by the user device and stored in the smart card. There is a local network wire or wireless that can be used for communication between user devices, smart home devices, and a local (edge) authentication server.

3.2 Use Case

In the smart home system, we have different IoT devices that provide various services to the user connected to gather using a local server that controls access to these devices. Users can access and controls these IoT devices from either inside or outside the home. In this case, we have two scenarios.

3.3 Smart System Access Scenarios

The first scenario is that the user wants to access the IoT device from inside the home using the local network. Therefore, there is a need for an offline authentication system to connect and access the smart home system. The second scenario is that the user wants to get access to the IoT device from outside the home using the internet connection. In this case, online authentication is needed to allow users to access the smart home system and control the IoT devices. In both scenarios, user privacy must be concerned. Anonymous access must be implemented to consider user privacy and comply with GDPR.

3.4 System Architecture

In any smart home system, several IoT devices act as smart home devices connected to the main server that controls access to these devices by authorized users. The IoT devices can connect to the management server using a local gateway, or edge server. The user can access the smart home service from outside or inside the home. As shown in Fig. 3. The IoT device in the smart home system needs a secure access mechanism using lightweight authentication according to its resource limitation in memory and processing. Authentication systems must be a concern to user privacy and be complies with GDPR. Efficient and has immunity to security attacks. So according to all these required features, we proposed a model designed according to the system in Fig. 4. It provides an Authentication method that can be used in online and offline scenarios. This system consists of:

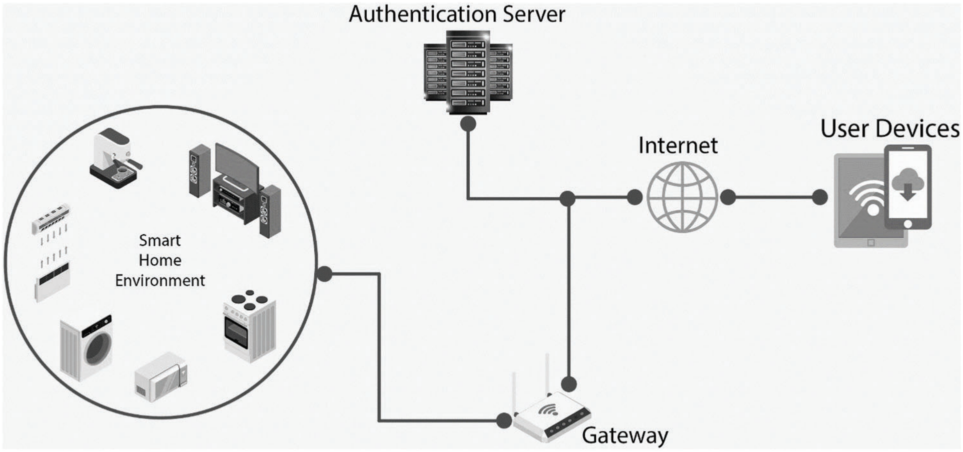


Figure 3: Smart home architecture

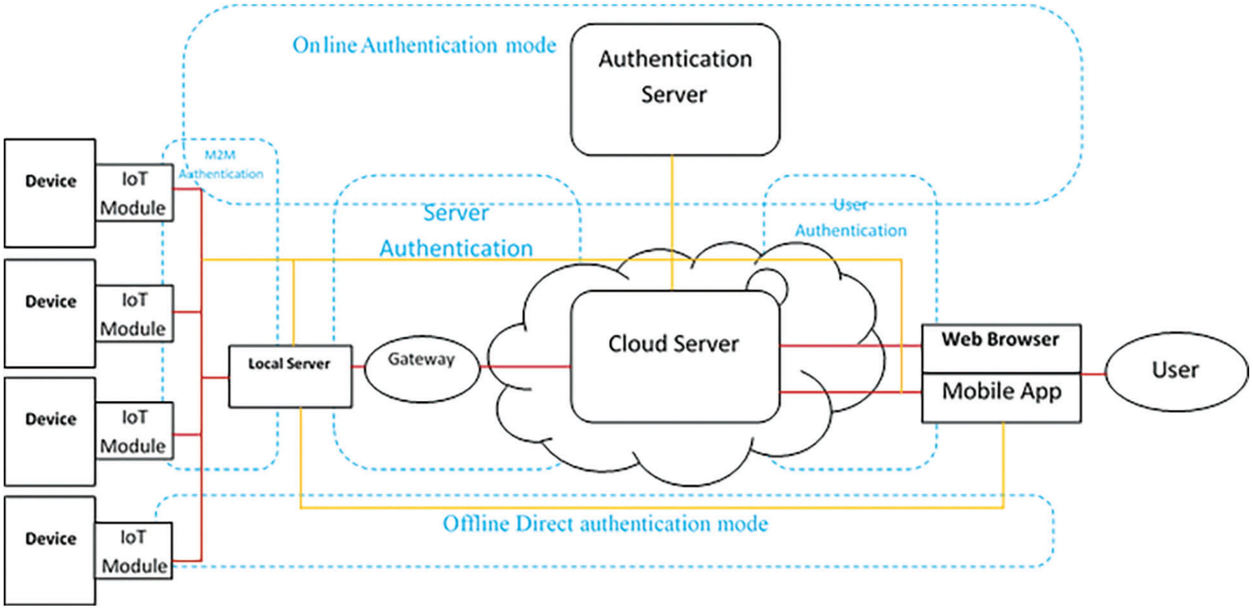


Figure 4: GDPR compatible smart home authentication system

- Authentication Level

This system, include four authentication levels

User authentication: this authentication between the user interface (mobile app or web page) and the cloud server.

Server authentication: between the cloud server and local (edge) server.

M2M authentication: between the local server and IoT devices.

Direct authentication: between the user (mobile app) and smart device for direct use in the offline environment mode.

- Authentication mode

  This system has two modes, first is an online mode using an authentication server to authenticate the user and smart home devices. The second is an offline mode for direct user authentication in case of no internet connection and offline environment.

- Device-level

  There are four device levels. First is the user device which is the device that is used by a user (mobile, Pc, etc.). The second is the authentication server which is the main online authentication server that is used to authenticate all system components. The third is the cloud server which provides the cloud service. The Fourth is the local (edge) server that is used to connect and manage the IoT devices.

  IoT devices: the smart home devices that are connected to the local server.

- Authentication mechanism

  Online mode: using the online authentication server for authenticating the users and devices in the network. This mode needs internet access for implementation.

  Offline mode: In this mode, the user and the devices need authentication without internet access, just using the local network and edge server.

3.5 Propose Module

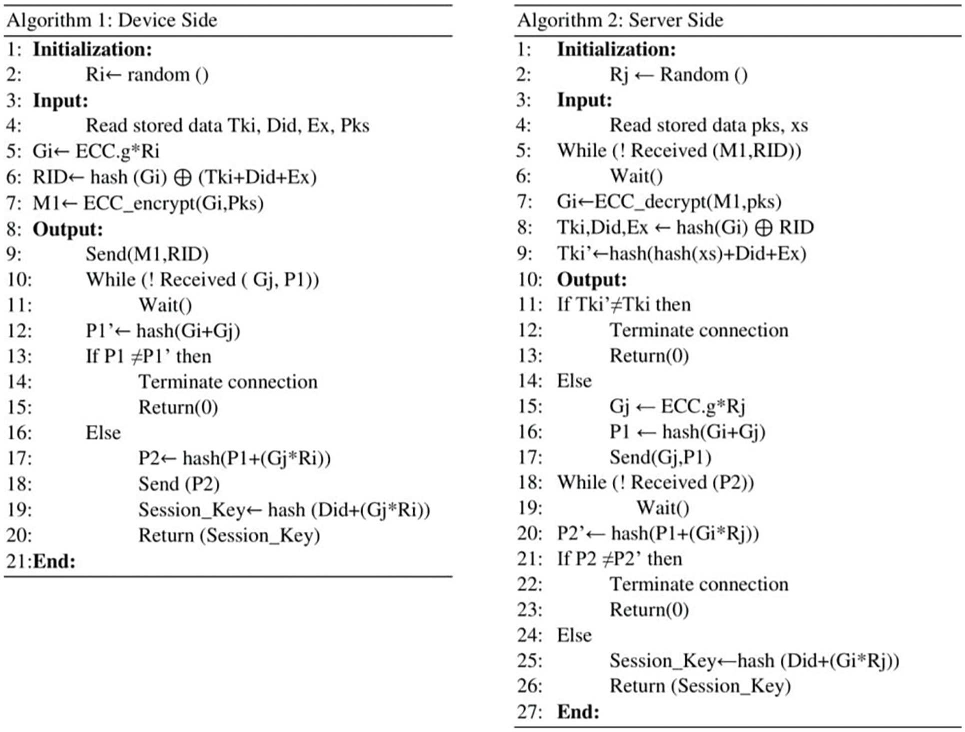
This study proposed an improvement to the protocol proposed by Wu et al. [13] in 2020. The original protocol needs a secure channel for the registration phase and there is no alternative way for authentication in the offline environment. Our scheme is the improved protocol with the local server to authenticate the user in the local network. Tab. 2. depicts all the notations and symbols that are used in the protocols and algorithms.

3.6 Proposed Algorithm

This study proposed the authentication algorithm for the user device, IoT device, and local server using pseudocode as shown in Fig. 5.

Table 2: The notation of the proposed protocol

Symbol	Significance
Sid_i ,Did_i ,Uid_i	Unique identification for the Server, IoT device and, User device
p_s	Local server private key
Pk_s, Pk_d	A local server, IoT device public
x_s	Local server long-term secret key
h(.)	One way hash function
g	ECC generation point (basepoint)
Rtm	Registration time
R_i, R_j	Random numbers
RID, Gi,Gj, P1,P2	Authentication messages
Enc,Dcr	Asymmetric key encrypting and decryption
Sk	Session key



```
Algorithm 1: Device Side
Initialization:
        Ri← random ()
Input:
        Read stored data Tki, Did, Ex, Pks
Gi← ECC.g*Ri
RID← hash (Gi) ⊕ (Tki+Did+Ex)
M1← ECC_encrypt(Gi,Pks)
Output:
        Send(M1,RID)
        While (! Received ( Gj, P1))
                Wait()
        P1'← hash(Gi+Gj)
        If P1 ≠P1' then
                Terminate connection
                Return(0)
        Else
                P2← hash(P1+(Gj*Ri))
                Send (P2)
                Session_Key← hash (Did+(Gj*Ri))
                Return (Session_Key)
End:
```

```
Algorithm 2: Server Side
Initialization:
        Rj ← Random ()
Input:
        Read stored data pks, xs
While (! Received (M1,RID))
        Wait()
Gi←ECC_decrypt(M1,pks)
Tki,Did,Ex ← hash(Gi) ⊕ RID
Tki'←hash(hash(xs)+Did+Ex)
Output:
If Tki'≠Tki then
        Terminate connection
        Return(0)
Else
        Gj ← ECC.g*Rj
        P1 ← hash(Gi+Gj)
        Send(Gj,P1)
While (! Received (P2))
          Wait()
P2'← hash(P1+(Gi*Rj))
If P2 ≠P2' then
        Terminate connection
        Return(0)
Else
        Session_Key←hash (Did+(Gi*Rj))
        Return (Session_Key)
End:
```

Figure 5: Proposed authentication algorithm for the device and the server

3.7 Installation Phase

For each device in this model (smart device, user device, or server) we assigned it with a universally unique ID (UUID) with 128-bit length compliant with RFC 4122 [22,23]. This UUID is generated base on SHA-1 of a namespace (ISO object ID) and SHA-1 of network MAC address. The server generates ECC key pair (Privet, Public) using brainpoolP256r1 256-bit prime field Weierstrass curve [24,25], and a long-term secret key (x) with 256-bit length.

3.8 Registration Phase

The proposed scheme uses the registration phase for the user device and IoT device.

3.8.1 User Device Registration Phase

In this phase, the user sends a registration request by presenting the smart card containing the UID to the server. The server also displays the QR code that contains the secrete token. Then the user device reads this QR code using the built-in camera. The QR code is shown for a limited time, and the local server clears the smart card data. The steps are shown in Fig. 6.

Step 1: The user sends the UIDi to the local server by the smart card.

Step 2: The local server computes the secrete token with an expiration date by this equation Tki = h(h(xs)|UIDi|Rtm). Also, store the UIDi in the server database.

Step 3: Generate QR code contain secrete token Tki, expiration date Ex, and the public key Pks. then show the QR code in the display.

Step 4: User device read the QR code and store Tki, Rtm, and Pks. to be used for the login authentication.

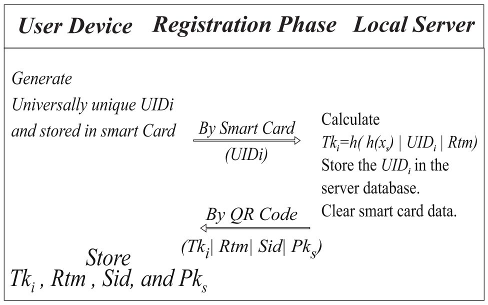


Figure 6: User device registration phase

3.8.2 Iot Smart Device Registration Phase

In this phase, the smart device requests registration and presenting the smart card containing the Did, and device public key to the server. The server generates the secrete token encrypted with the device's public key. Then the smart device receives the token by WiFi or BLE channel. The steps are shown in Fig. 7.

Step 1: The IoT Device sends the Uidi, Pkd to the local server by the smart card.

Step 2: The local server compute the secrete token with an expiration date by this equation

Tki = h(h(xs)|Didi|Rtm). Also, store the Didi and Pkd in the server database.

Step 3: Generate a message contain secrete token Tki, expiration date Ex, Server Sid, and the public key Pks. Encrypted with IoT device public key Pkd then send the message to the IoT device through BLE or WiFi Channel.

Step 4: IoT device receives the encrypted message and stores Tki, Rtm, and Pks. after decrypting it, to be used for login authentication.

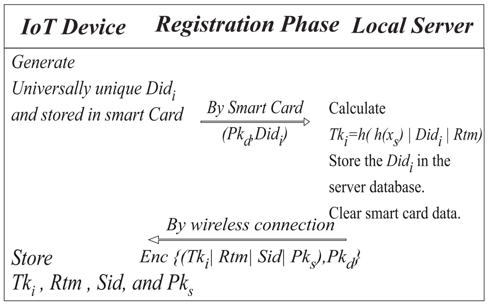


Figure 7: IoT smart device registration phase

3.9 Authentication Phase

The proposed scheme uses the authentication phase for the user device and IoT device.

3.9.1 User Device Authentication Phase

In this phase, the user device stored secrete token Tki, the expiration date of the token Rtm, and the public key of the server Pks. The following steps describe the login authentication and session key establishment. As shown in Fig. 8.

Step 1: User device choose random number Ri to calculate Gi = g*Ri, and RIDi = h(Gi) ⊕ (Tki |UIDi| Rtm).

Step 2: User device create a message with Gi encrypted by server public key Pks, and RIDi to the server

Step 3: Server decrypted Gi using private key *ps* then calculate (Tki|UIDi|Rtm) = h(Gi) ⊕ RIDi

Check the UIDi in the database, then calculate Tki` = h(h(xs)|UIDi|Rtm).

Check Tki`? = Tki if Tki` ≠ Tki then authentication process will terminate, otherwise

Choose random number Rj, Calculate Gj = g*Rj , P1 = h(Gi|Gj). And send Gj, P1 to the user device.

Step 4: User device receives the Gj, P1 and calculates P1` = h(Gi|Gj).

Check P1`? = P1 if P1` ≠ P1 then authentication process will terminate, otherwise

Calculate P2 = h(P1|Gj*Ri), then send P2 to the local server.

Step 5: Local server receive P2, and calculate P2` = h(P1|Gi*Rj) since Gi*Rj = Gj*Ri = g*Ri*Rj.

Check P2`? = P2 if P2` ≠ P2 then authentication process will terminate, otherwise

Mutual Authentication is established, and the session key is Sk = h(UIDi|g*Ri*Rj).

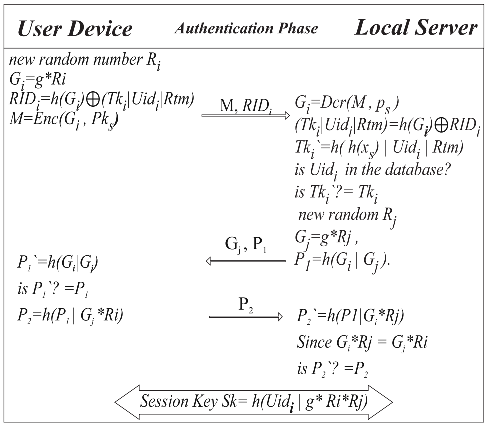


Figure 8: User device authentication phase

3.9.2 Iot Device Login Authentication Phase

In this phase, the user device stored secrete token Tki, the expiration date of the token Ex, and the public key of the server Pks. The following steps describe the login authentication and session key establishment. As shown in Fig. 9.

Step 1: IoT device choose random number R_i to calculate $Gi = g*Ri$, and $RID_i = h(G_i) \oplus$ (Tk$_i$|UID_i| Rtm).

Step 2: IoT device creates a message with *Gi* encrypted by server public key Pk_s, and RID_i to the server

Step 3: The local server decrypted *Gi* using private key p_s then calculate $(Tk_i\,|UID_i|Rtm) = h(G_i) \oplus RID_i$

Check the UID_i in the database, then calculate $Tk_i` = h(\,h(x_s)|UID_i|Rtm)$

Check $Tk_i`? = Tk_i$ *if* $Tk_i` \neq Tk_i$ then authentication process will terminate, otherwise Choose random number R$_j$, Calculate $Gj = gRj$, $P_1 = h(G_i|G_j\,)$. And send G_j, P_1 to the smart device.

Step 4: IoT smart device receives the G_j, P_1 and calculates $P_1` = h(G_i|G_j)$

Check $P_1`? = P_1$ *if* $P_1` \neq P_1$ then authentication process will terminate, otherwise

Calculate $P_2 = h(P_1|Gj*Ri)$, then send P_2 to the local server.

Step 5: Local server receive P$_2$, and calculate $P_2` = h(P1|Gi*Rj)$ since Gi*Rj = Gj*Ri = g*Ri*Rj

Check $P_2`? = P_2$ if $P_2` \neq P_2$ then authentication process will terminate, otherwise Mutual Authentication is established, and the session key is $Sk = h(UID_i|g{*}Ri{*}Rj)$.

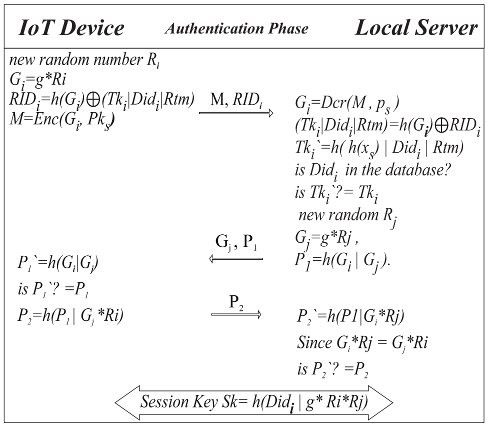


Figure 9: IoT device login authentication phase

4 Security Analysis and Formal Verification

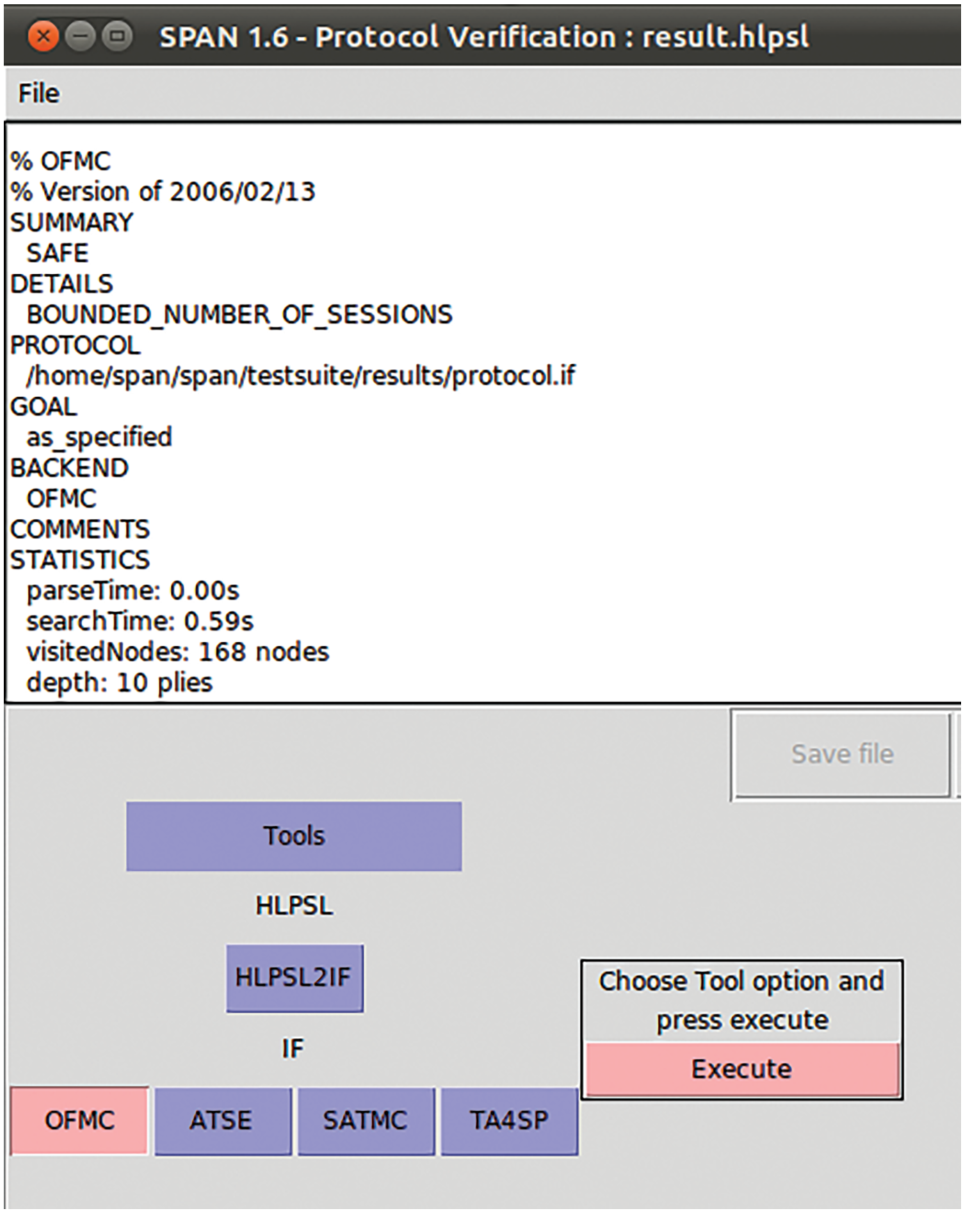
4.1 Security Analysis

- **Secure Channel**: The proposed model defines the secure channel for registration since it is using smart cards and dynamic QR code encryption. All secrete data are encrypted and securely transmitted through the communication channels.
- **Resist Man-In-The-Middle Attacks**: An attacker cannot implement attacks via the visual Out Of Band (OOB) channel. In addition to the encrypted messages that are transmitted in the communication channel. The session key is exchanged securely, so the attacker cannot acquire it.
- **Resist Denial of service (DoS) Attacks**: The proposed model can resist DoS attacks by blocking the attacker packets after a specific number of failed authentication attempts using an intruder detection system. Also using edge server as an alternative to manage the locally connected device.
- **Resist Reply Attacks**: This study is based on randomly generated secret numbers and different nonce in each authentication attempt for message verification and exchange a new session key to preventing the attacker from reuse an old message.
- **Resist Eavesdropping Attacks**: All transmitted data is encrypted using symmetric or asymmetric encryption, to securely exchange session keys. Therefore, the attacker cannot obtain any useful information from the transmitted messages.
- **Resist Key Logging Attacks**: There is no input for any secrete authentication data (like passwords) using an input device for each authentication attempt.
- **Data Secrecy and Reliability**: The data are transmitted in an encrypted and signed form for a secure session key exchange. all transmitted data after authentication are verified and encrypted using the session key.
- **Privacy Consideration**: Generating different user IDs in every login authentication attempt based on the random number and secrete token. The user identification is untraceable, and anonymity is considered.

- **Forward Secrecy**: a new session key is generated for every new session. And it will be useless in the next session. Every session key is created based on a randomly generated number and securely exchanged. There is no way to use the old session key to recalculate or expect the next session key.

4.2 Formal Verification Using AVISPA

Different types of protocol verification tools are used to check and verify the protocols; one of the most well-known for formal verification is (AVISPA). This tool supports the (HLPSI) high-level protocol specification language. In this work, we checked our protocol using the AVISPA tool to verify whether our proposed protocol suffers from security attacks. The result is safe according to the specified goals, as shown in Fig. 10.



```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/protocol.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.59s
  visitedNodes: 168 nodes
  depth: 10 plies
```

SPAN 1.6 - Protocol Verification : result.hlpsl
File
Save file
Tools
HLPSL
HLPSL2IF
IF
OFMC
ATSE
SATMC
TA4SP
Choose Tool option and press execute
Execute

Figure 10: AVISPA SPAN protocol check result

5 Performance Evaluation

This section discusses efficiency evaluation and compression with similar works.

5.1 *Algorithm Time Complexity*

In the algorithm that proposed in Section 5 of this study, we analyze the algorithm in terms of Big O notation. The algorithm divided according to the calculation complexity of the functions that are used in this algorithm. The study used Encryption and Decryption with $O(n^2)$, the one-time hash function, Random number generation, and bitwise operation with $O(n)$, and some tiny fixed calculation operation with $O(1)$. There is no complex operation and extreme nested loops that are used with very High O notation in the algorithm. All the calculations and process operations are lightweight and suitable for resource-constrained devices.

5.2 *Computational Complexity Cost*

The study compared the computation complexity and cost between the proposed model and related existing models in Wang et al. [12], Kumar et al. [21], and Wu et al. [13]. **Tex, Tmp, Tmu, Tbio, Th, Tx,** and **Ted** denote the execution time of Modular exponential of the ECC point, bilinear mapping, scalar multiplication of the ECC point, bio hashing time, one direction hash function, XoR operation, and symmetric encryption/decryption. Tab. 3. Contains the result of the computation complexity comparison.

After calculating the computational complexity cost and process speed. The study founds that Wang et al. [12] takes the highest computation cost. Wu et al. [13] also need more computation time because they used modular exponential and bilinear mapping, which time-consuming operation and higher computation complexity. Kumar et al. [21] are based on a simple mathematical operation that shows the lowest time consumption at the expense of the security level. The proposed model with acceptable computation cost is needed. Because it uses efficient functions, the highest security level, and low complexity mathematical operation.

5.3 *Security Property*

The study presented the security property shown in Tab. 4. Between the proposed model and related existing models in [12,13,21]. In contrast with these models, our model has more security properties and prevents more security attacks.

According to the table, the study found that Wang et al. [12] model is insecure against different types of security threats, Wu et al. [13] is more secure but still exposed to some security threats. Kumar et al. [21] also insecure against many attacks type and some of them are not considered. The proposed model is more secure against different types of security threats Like DoS which other similar models are not. In our model, it considers privacy, anonymity, data secrecy and reliability, forward secrecy, and GDPR Compliance, while the compared model are not supported these security features.

5.4 *Message Size and Communication Cost*

The study compared the communication cost between the proposed model and related existing models in [12,13,21], according to the message size. Tab. 5. Show the communication cost result in message number and message size.

Table 3: Computational complexity costs

Model	Computational complexity	Cost
Wang et al. [12]	8Tmu + 11Th + 1Tx	209.0221 ms
Kumar et al. [21]	10Ted + 4Tbio + 7Th	202.5000 ms
Wu et al. [13]	2Tex + 4Tmu + 1Tmp + 10Th + 2Tx	164.5702 ms
Proposed model	6Tmu + 13Th + 2Tx + 2Ted	156.9636 ms

Table 4: Security property

Security property	Wang et al. [12]	Kumar et al. [21]	Wu et al. [13]	Proposed model
Man-in-the-middle attacks	Yes	Yes	Yes	Yes
Reply attacks	Yes	Yes	Yes	Yes
Denial of service (DoS)	No	Yes	No	Yes
Eavesdropping attacks	No	Yes	Yes	Yes
Data secrecy and reliability	No	Yes	No	Yes
Privacy consideration	No	No	No	Yes
Anonymity	No	yes	No	Yes
Mutual authentication	Yes	Yes	Yes	Yes
Forward secrecy	No	Yes	No	Yes
GDPR compliance	No	No	No	Yes

Table 5: Message size and communication cost

Model	Massage size	Number of messages
Wang et al. [12]	1568 bit	3
Kumar et al. [21]	959 bit	5
Wu et al. [13]	1404 bit	3
Proposed model	1440 bit	3

6 Conclusion

This study proposed a new model, benefit from previous methods' strength points and overcome the weakness, by combining different techniques and extracting a new approach with better performance and more security. Our scheme draws on a variety of currently popular technologies. Our scheme used QR Code as a visual channel, smart card to send the unique ID. It is based on a lightweight process and fast mathematical operations to be suitable for resource-constrained devices. Our scheme does not assume the secure channel existence. It can work and authenticate devices indoors in an offline environment using a local authentication server. Resist different types of security threats. Similar works need more time consumption, more communication cost, and they have low-security features. Some of them need low communication costs at the expense of the security level. Our proposed protocol has more security features and compliance with GDPR. It has the lowest calculation time consumption and communication overhead cost for the required security level comparing with other similar works.

For future works, we are planning to improve the performance and communication cost according to our measurement and decrease the usage of resources. And we will implement the experimental model for our proposed schemes in the IoT environment system. Software-defined networks (SDN) and Machine Learning (ML) could be involved in this model.

Funding Statement: The authors received no specific funding for this study.

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

References

[1] M. El-hajj, A. Fadlallah, M. Chamoun and A. Serhrouchni, “A survey of internet of things (IoT) authentication schemes,” *Sensors,* vol. 19, no. 5, pp. 1141, 2019.

[2] L. Atzori, A. Iera and G. Morabito, “The internet of things: A survey,” *Computer Networks,* vol. 54, no. 15, pp. 2787–2805, 2010.

[3] A. K. Singh, M. Alshehri, S. Bhushan, M. Kumar, O. Alfarraj *et al.*, “Secure and energy efficient data transmission model for WSN,” *Intelligent Automation & Soft Computing,* vol. 25, no. 3, pp. 761–769, 2021.

[4] M. Drev and B. Delak, “Conceptual model of privacy by design,” *Journal of Computer Information Systems,* vol. 61, pp. 1–8, 2021.

[5] C. J. Hoofnagle, B. V. D. Sloot and F. Z. Borgesius, “The european union general data protection,” *Information & Communications Technology Law,* vol. 28, no. 1, pp. 65–98, 2019.

[6] C. Tikkinen-Piri, A. Rohunen and J. Markkula, “Eu general data protection regulation: Changes and implications for personal data collecting companies,” *Computer Law & Security Review,* vol. 34, no. 1, pp. 134–153, 2018.

[7] N. N. Loideain and R. Adams, “From alexa to siri and the GDPR: The gendering of virtual personal assistants and the role of data protection impact assessments,” *Computer Law & Security Review,* vol. 36, pp. 105366, 2020.

[8] F. Hussain, R. Hussain, B. Noye and S. Sharieh, “Enterprise API security and GDPR compliance: Design and implementation perspective,” *IT Professional,* vol. 22, no. 5, pp. 81–89, 2020.

[9] M. El-hajj, M. Chamoun, A. Fadlallah and A. Serhrouchni, “Taxonomy ofauthentication techniques in internet of things (IoT),” in *2017 IEEE 15th Student Conf. on Research and Development (SCOReD)*, Wilayah Persekutuan Putrajaya, Malaysia, 2017.

[10] S. Kalra and S. K. Sood, “Secure authentication scheme for iot and cloud servers,” *Pervasive and Mobile Computing,* vol. 24, pp. 210–223, 2015.

[11] C. Chang, H. Wu and C. Sun, “Notes on “secure authentication scheme for iot and cloud servers”,” *Pervasive and Mobile Computing,* vol. 38, pp. 275–278, 2017.

[12] K. Wang, C. Chen, W. Fang and T. Wu, “A secure authentication scheme for internet of things,” *Pervasive and Mobile Computing,* vol. 42, pp. 15–26, 2017.

[13] H. Wu, C. Chang and L. Chen, “Secure and anonymous authentication scheme for the internet of things with pairing,” *Pervasive and Mobile Computing,* vol. 67, pp. 101177, 2020.

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

[15] B. Yu and H. Li, “Anonymous authentication key agreement scheme with pairing-based cryptography for home-based multi-sensor internet of things,” *International Journal of Distributed Sensor Networks,* vol. 15, no. 9, pp. 1550147719879379, 2019.

[16] S. Naoui, M. h. Elhdhili and L. A. Saidane, "Novel smart home authentication protocol lrp-shap," in *IEEE Wireless Communications and Networking Conf. (WCNC)*, Marrakesh, Morocco, 2019.

[17] M. Hussain and U. Jain., "Simple and secure device authentication mechanism for smart environments using internet of things devices," *International Jornal of Communication System,* vol. 33*,* no. 16*,* pp. e4570, 2020.

[18] Q. Lyu, N. Zheng, H. Liu, C. Gao, S. Chen *et al.*, "Remotely access "my" smart home in private: An anti-tracking authentication and key agreement scheme," *IEEE Access,* vol. 7*,* pp. 41835–41851, 2019.

[19] U. Coruh and O. Bayat, "Hybrid secure authentication and key exchange scheme for m2m home networks," *Security and Communication Networks,* vol. 2018*,* pp. 6563089, 2018.

[20] X. Sun, S. Men, C. Zhao and Z. Zhou, "A security authentication scheme in machine-to-machine home network service," *Security and Communication Networks,* vol. 8*,* no. 16*,* pp. 2678–2686, 2015.

[21] P. Kumar and L. Chouhan, "A secure authentication scheme for iot application in smart home," *Peer-to-Peer Networking and Applications,* vol. 14*,* pp. 420–438, 2020.

[22] S. Putz, T. Wiemann and J. Hertzberg, "The mesh tools package – introducing annotated 3d triangle maps in ros," *Robotics and Autonomous Systems,* vol. 138*,* pp. 103688, 2021.

[23] A. J. Poulter, S. J. Ossont and S. J. Cox, "Enabling the secure use of dynamic identity for the internet of things—using the secure remote update protocol (SRUP)," *Future Internet,* vol. 12*,*.no. 8*,* pp. 138, 2020.

[24] N. H. Kumar and G. Deepak, "Mutual authentication and data security in iot using hybrid mac id and elliptical curve cryptography," *Turkish Journal of Computer and Mathematics Education (TURCOMAT),* vol. 12*,* no. 11*,* pp. 501–507, 2021.

[25] B. R. Harsha, A. Damodaran, S. Ranganath, V. Raut and S. Holla, "An approach to enable secure and reliable communication on IoT devices," in *2019 4th Int. Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, Bengaluru, India, 2019.