A Strong Mutual Authentication Protocol for Securing Wearable Smart Textile Applications

Hakan DALKILIÇ, Mehmet Hilal ÖZCANHAN
Department of Computer Engineering, Faculty of Engineering
Dokuz Eylül University, Izmir, 35390, Turkey
dalkilic.hakan@ogr.deu.edu.tr

***Abstract*—With increasing modern technology involvement in numerous consumer areas, our cities are gradually turning into smart urban areas. Wireless technologies have especially been playing a pivotal role in making cities smarter. The popular name for wireless wearable devices is the wearable Internet of Things (IoT). Wearable IoT has begun a smart textiles movement. However, wearable IoT increased wirelessly transmitted data, opening avenues for critical data capture by unauthorized listeners. The present study offers a typical wearable textile IoT device with information security. Our work proposes a novel mutual authentication protocol between IoT devices and their gateway, supported by a state-of-the-art encryption algorithm. The protocol can increase the information security of similar smart textiles. In addition to an informal security evaluation, our protocol has been tested by two formal security analysis tools. The popular Scyther and AVISPA tools verify that the data transmission between our design wearable textile and the gateway is secure. A comparison of our work with previous proposals shows the comprehensiveness of our design and its applicability to other IoT devices, as well.**

***Index Terms*—authentication, cryptography, Internet of Things, message authentication, wearable sensors.**

I. Introduction

The concept of the Internet of Things (IoT) was first proposed with the rise of radio frequency identification (RFID) technology in 1999 [1]. IoT allows communication and data sharing easily and quickly [2-3]. The ability of small devices to communicate wirelessly with other large devices rapidly increased consumers' mobility in the world. The conveniences provided by IoT in healthcare, transportation, and other daily life activities increased the number of smart cities [4]. State-of-the-art technologies support smart cities by providing numerous facilities for the daily life of their dwellers [5-6]. Mobile phones and wearable smart devices play an essential role in the critical services of a smart city [7-8]. Wearable devices are finding their way into many application areas, among which smart textiles are increasing. The wearable devices-textiles merger can help retrieve personal health information, making them indispensable in smart cities.

As IoT devices worldwide reach billions, critical tasks face many information capture attacks every day [9]. The attackers seek to gain access or alter data, resulting in depletion of factual data and creation of false and erroneous information. Some other attacks aim at depleting the energy of the un-resourceful IoT devices. Attacks can cause the services of wearable smart textile devices to be temporarily or permanently blocked, with eventual endangerment of their users. The proper operation of the smart city systems can only be guaranteed by providing security to wearable devices. Therefore, a great deal of research is currently being conducted in IoT security [10]. Since there are numerous types of attacks on IoT, sophisticated precautions are necessary to prevent most of them in one go. Due to IoT devices' low energy and memory space, countermeasures against IoT attacks differ from countermeasures for computers [11-12]. Computer countermeasures are resource-demanding and more sophisticated. Nevertheless, IoT information security is possible as long as the security steps are optimum and the system performance is not affected.

Authentication is a security service that plays an integral part in IoT devices and computers [13-16]. Authentication is the process of verifying the other party's identifier. Mutual authentication is the process of both parties verifying each other's identifiers. Wearable smart textile devices need a strong mutual authentication protocol for a reliable decision, which can only be based on accurate information. In the present work, we propose a design that provides a secure mutual authentication protocol to prevent potential unauthorized access to the data between wearable smart textile devices and their IoT gateway. The rest of the article includes Section 2 for works related to our study and our motivation and contribution. Section 3 contains our material and methods, while Section 4 describes the security analysis of our proposed protocol. Section 5 presents the security performance comparison of our work against previous works. Finally, Section 6 gives the conclusion of our work.

II. Related Works

There are numerous studies on the security of IoT devices, which follow the research in RFID security. Some of the studies related to our work are summarized below.

An IoT system using biometric technology is proposed in a comprehensive work [17]. In the work, where data encryption using the advanced encryption standard (AES) algorithm is proposed for securing the information exchange in the system, Raspberry Pi is used as a resourceful IoT device. The encrypted biometric data is hosted on the Azure cloud.

Sun et al. have designed a password-based authentication protocol in the machine-to-machine (M2M) data transfer between IoT members [18]. The study presents a design to eliminate impersonation and replay attacks. In the proposed design, although a mutual authentication is proposed between the mobile devices and an M2M server, the authentication between gateway and home devices is not considered. A security analysis with a formal verification

Digital Object Identifier 10.4316/AECE.2022.01004

tool is also lacking. Aydın et al. developed an authentication protocol for lightweight devices, i.e., devices with scarce resources [19]. Their protocol also makes use of the AES algorithm. The recommended protocol is secure against de-synchronization and denial-of-service (DoS) attacks.

A smart home system was designed using IoT devices at work [20]. A smart home is an indispensable part of IoT smart systems, which constitutes smart cities. The low-cost system design controls the utensils of a house from anywhere on earth. A three-level Kerberos authentication has been used in the work as a precaution against security problems. The work has drawn many references in the literature, as Kerberos is not accepted as an effective solution for authentication [10].

Some specific IoT attack types such as side-channel attacks, signal injection, spoofing, elevation of privilege, privacy breach, information disclosure, device tampering, and DoS attacks have been studied in work [21]. Security and privacy concerns in IoT systems were addressed, but protocol-specific threats were not considered in the article [22].

Multiple authentication protocols have been proposed in works [23-26], using physically unclonable functions (PUFs), long-term evolution (LTE) direct technology, wireless sensor network (WSN) authentication schemes, fog computing nodes. All of the works aim at perfecting the security and privacy of information exchange in IoT systems. Nikooghadam and Amintoosi [27] suggested a notable authentication scheme for medical information systems. The scheme provides a mutual authentication protocol and resists key compromise by impersonation, password guessing, replay, and insider attacks. However, the proposed scheme is not evaluated for known passive attacks.

In 2006, Chang and Le proposed an authentication scheme for Ad hoc wireless sensor networks, ensuring forward secrecy [28]. Nevertheless, Das et al. [29] showed that the protocol is insecure and insufficient in authentication and password update phases. Another authentication scheme was also attacked [30]. The authors claimed that their protocol was verified by the Scyther tool. Although the authors claimed that the proposed mutual authentication protocol provides security for man-in-the-middle, DoS, and replay attacks, work [31] demonstrated that an attacker can capture the certification from the initialization message of the authentication. The attack results in impersonation of the base station by using the captured certification and some other fake parameters.

There are also proposals for wearable sensors in the literature [32-33]. The lightweight authentication protocol proposed in work [32] is verified using the Automated Validation of Internet Security Protocols and Applications (AVISPA) tool. In work [33], the collected inertial data is sent to a cloud server wirelessly. However, the sensor data is sent without any security measures. Similarly, the wireless device in our previous work sends session patient arm flexion and extension angles to a gateway in cleartext [34]. We witnessed that the sent data becomes critical when a malicious user intercepts it and understands that the arm of the tracked patient is not functioning normally. The above threat led us to provide security to our wearable device in the present work.

A. Motivation and Contributions

The above literature review shows that information security must be considered continuously against the risk of new eavesdropping and attack techniques on wireless sensor data. For example, our previous work showed a massive vulnerability because the patient data was sent in cleartext [34]. The detected vulnerability was the cause of our primary motivation for designing two-factor security for wearable health sensors, especially after the "bluejacking" and "bluesnarfing" attacks revealed in work [35]. The mentioned attacks allow hackers to expose the information transmitted in Bluetooth communication. With such motivation, we aim to contribute the following in our present work:

- A formally analyzed (by AVISPA and Scyther tools) authentication protocol to ensure secure information exchange by IoT devices;
- Addition of secure wearable sensors on textiles for the creation of smart textiles;
- A novel wearable sensor application for smart textiles, smart healthcare, and smart cities.

In brief, we intend to contribute by presenting a secure smart wearable device for a novel smart textile prototype that can be used in healthcare, making smart cities more intelligent. The next chapter details our material and methods.

III. MATERIALS AND METHODS

A. Materials

This section explains the materials used to create a wearable device that makes the worn textiles smart. The resulting smart textile performs the human elbow flexion and extension angle measurements. An authentication protocol is added to ensure the secure transmission of the measured angles in the design.

The smart textile device has been created by integrating sensors with a microcontroller card attached to the textile with conductive wires present inside the woven fabric. The materials used in the proposed system are as follows:

- Arduino UNO R3 microcontroller card (sensor & communication base);
- MPU 6050 accelerometer and gyro sensor (2 units);
- HC-05 Bluetooth module;
- Raspberry Pi 4 (IoT gateway);
- Passive elements & conducting wire.

Two units of MPU 6050 are used to measure the elbow flexion and extension angle θ, in Figure 1. Both MPU 6050 sensors are fixed on the textile. The data received from the sensors are processed using an Arduino UNO microcontroller card and stored after being encrypted with AES. Afterward, it is transmitted wirelessly to the Raspberry Pi device (gateway) from the Arduino UNO using the HC-05 Bluetooth module. Conductive wires of the textile are used in the communication of the sensors with Arduino UNO. In addition, conductive wires are used to position the sensors and Arduino UNO on the textile. The materials used in our prototype are shown in Figure 1 and represent an application of smart textiles on the human arm.

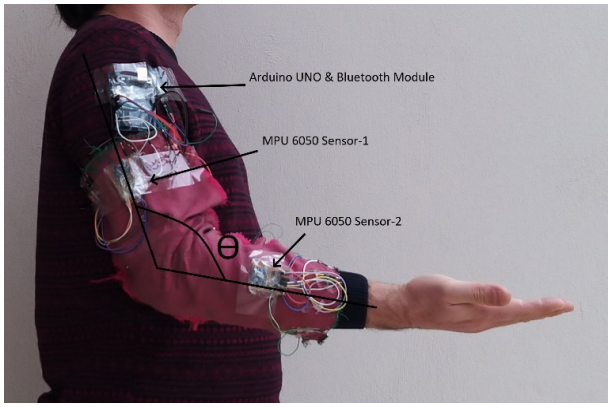


Figure 1. The prototype of wearable sensor applied on a human arm

After the sensor data collected are transmitted to the IoT gateway, they are presented to the user with a custom Windows application, as in Figure 2. The maximum and minimum values during the measurement are determined and displayed on the screen.

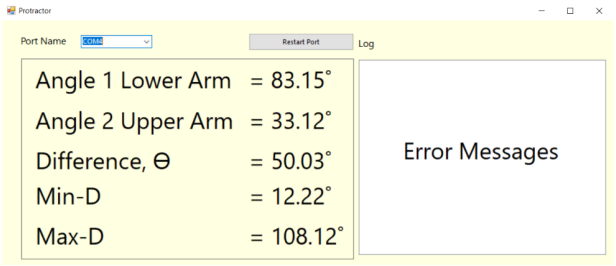


Figure 2. Application for measuring elbow flexion and extension angles

The Arduino UNO card is programmed with Arduino integrated development environment (IDE) software to collect and encrypt the data from the sensors. AES is used for encryption and an HC-05 Bluetooth module to transmit the data to the gateway.

B. *Methods*

One of the security services necessary is data confidentiality, which means the data collected and stored on the IoT device is not revealed to unauthorized persons. Confidentiality is provided through encryption algorithms [36]. Another critical security service is data integrity, i.e., the contents of the transferred data must be intact and unchanged [37]. Authentication protocols can ensure data integrity. Availability, another security service, ensures that access to data is possible only by authorized users. Availability can be achieved by preventing continuous unauthorized access attempts to IoT devices. Our recommended authentication protocol provides confidentiality, integrity, and availability.

After getting cautioned about the information security vulnerability in our designed prototype, we sought to add a mutual authentication protocol for providing the above-explained security services. We devised the protocol shown in Figure 3 between our smart textiles and its gateway. Then, we performed both informal and formal security tests needed to verify our protocol design. Thus, information security has been ensured against attacks via unauthorized Bluetooth users. However, it is assumed that the data communication between the IoT gateway and the cloud servers is secure. Thus, data communication between Arduino UNO and users is provided with end-to-end security. Next, the designed security authentication protocol is presented in detail.

1) *Setup of Our Secured Smart Textile Design*

In Figure 3, the setup of our smart textile configuration is shown. Data collected from sensors is transferred to Arduino UNO in cleartext. Data on Arduino UNO is encrypted with AES and stored. It is then transmitted wirelessly to the IoT gateway via Bluetooth. A mutual authentication protocol prevents unauthorized intervention to the wireless communication between the wireless IoT device and the gateway.

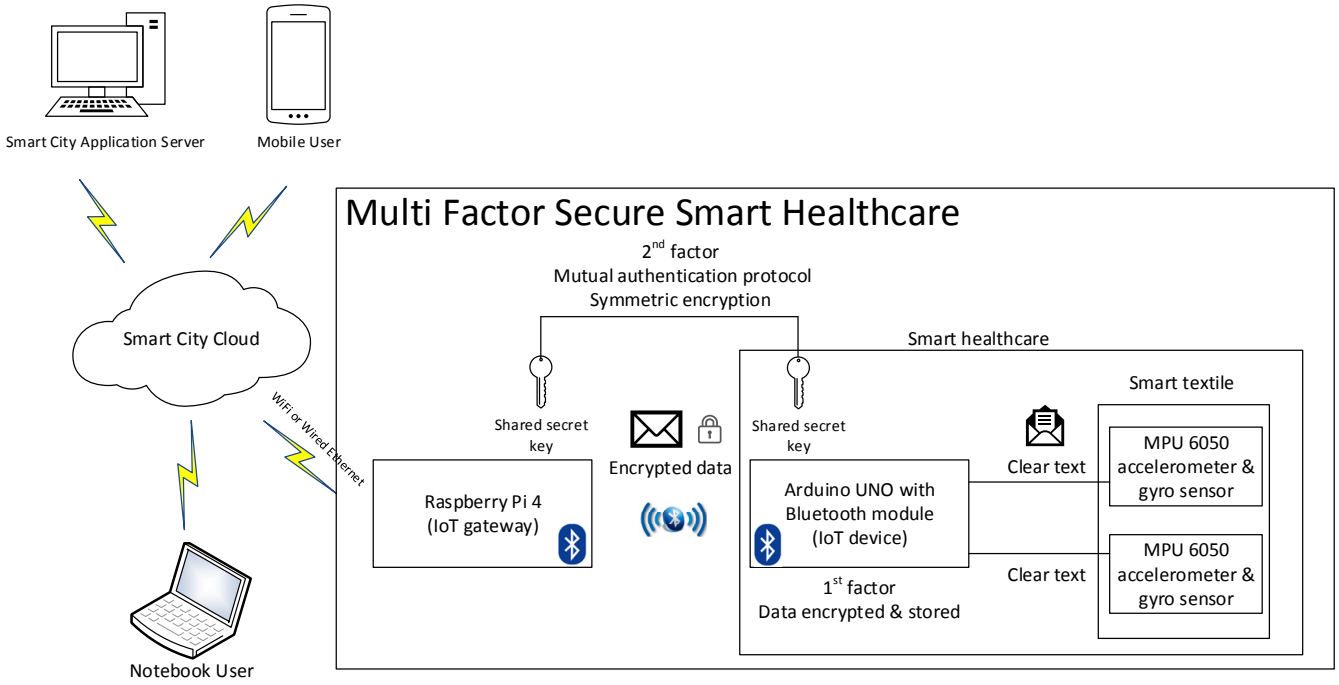
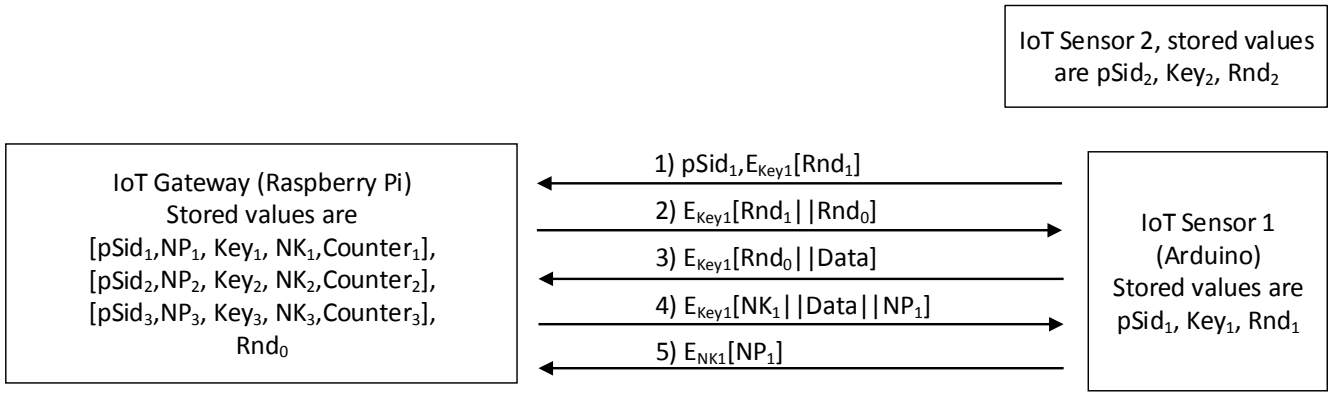


Figure 3. Setup of smart textile-secured smart healthcare-smart city



Abbreviations and notations used:

E	: Encryption with present day popular encryption algorithm (AES)
$pSid_1$	: Pseudo ID of the IoT sensor 1
Key_1	: Secret key for AES encryption algorithm
Rnd_0	: Random number of IoT gateway
$Counter_1$	: Counter for ID checking, counter value can be maximum 3
||	: Concatenation
Rnd_1	: Random number of IoT sensor 1
NK_1	: Secret key for AES encryption algorithm for the next round
NP_1	: Pseudo ID of the IoT sensor 1 for the next round

Figure 4. Our proposed novel mutual authentication scheme

By preventing any intervention from the start of the communication, the confidentiality-integrity-availability (CIA triad) of the smart textile systems is obtained. The designed mutual authentication protocol is explained in detail in the following subsection.

2) Our Novel Authentication Protocol for IoT Devices

Figure 4 exhibits the mutual authentication to be performed between IoT devices and their gateway. The abbreviations and notations used in the protocol are also shown.

The sensor ID is updated by the gateway in each authentication session to prevent the use of sensor IDs by impersonation attacks. Additionally, the popular encryption algorithm (AES) ensures the confidentiality of the data transmitted in the exchange. The initial value of the first session's secret key used is pre-embedded in the devices at the production stage. The gateway generates a new session key in each round, dictated to the wearable IoT device. A counter is kept for each sensor on the gateway in case the fourth and/or the fifth communication steps are interrupted. The counter value is initially set to 0 for each sensor. NP_1 initial value is '12345'. The steps of our mutual authentication are described below.

Step 1: First, the wearable IoT device sends its pseudo-ID ($pSid_1$) as plaintext, and the generated then encrypted pseudo-random number (Rnd_1) to the IoT gateway. The gateway locates the device in its database using $pSid_1$ and decrypts the message to obtain Rnd_1, using the sensor's secret key in the database.

The gateway will check both $pSid_1$ and NP_1 in step 1, as it may not have received the approval message in step 5 of the previous round. If the device tries to communicate with an old $pSid_1$, instead of NP_1, the counter is incremented by 1. If the counter value exceeds 3, the gateway detects an attack and disables the IoT device with $pSid_1$ value in its database.

Step 2: The IoT gateway returns the pseudo-random number it has received from the sensor (Rnd_1) after concatenating it with the pseudo-random number it has produced (Rnd_0). However, the message is encrypted using the secret key Key_1 found in the database across the corresponding sensor. The wearable IoT device decrypts the message and verifies its random number. Thus, the sensor authenticates the gateway.

Step 3: We assume that the gateway is authenticating one device at a time. Hence, the sensor sends the encrypted concatenation of the gateway's random number with the sensor data. The gateway decrypts the message and confirms it is coming from the same $pSid_1$ using Rnd_0. Only then, it trusts the sensor and write the data to its database. Thus, mutual authentication is completed.

Step 4: The gateway creates a new secret key (NK_1) and a new pseudo-ID (NP_1). Then, the newly created key, the received data, and the new pseudo-ID are concatenated and then encrypted before being sent to the sensor. The new values are recorded in the database. The sensor decrypts the message and checks if the data was received correctly. Upon verifying correct data transfer, the sensor device updates its session key with the new secret key (NK_1) and records the new pseudo-ID (NP_1).

Step 5: In the last step, the sensor sends the NP_1 value back to the IoT gateway encrypted with the new key NK_1. When the IoT gateway receives the message, it ensures that the sensor has updated its key and ID values. Finally, the IoT gateway resets the counter value of the related sensor.

3) AES Implementation of Our Work

AES encryption library for Arduino UNO shared on (https://github.com/DavyLandman/AESLib - accessed July 1, 2021) is used to secure the transmitted data. The same library has been used in other works [38]. The Arduino "micros ()" instruction tool was used for measuring the time spent in a process. Tables I and II summarize the resources

used during encryption and decryption by the IoT device. The Flash memory footprint of our five-step protocol on the IoT device is 4.67 kB. The SRAM memory load is only 0.21 KB when the protocol is run.

A thousand sequential encryptions have been carried out to measure the encryption latency. Dividing the total time by 1000, the average time spent for single encryption was calculated to be 282 microseconds (μs). Likewise, the average time spent for decryption was found to be 322 μs. To measure the total time taken by the IoT device to finish a full protocol round (3 encryptions and 2 decryptions), the protocol code was stripped of wireless communication steps. The total time for completing three encryptions and two decryptions was 1492 μs. This value agrees with the addition of individual encryption and decryption times, except the extra overhead 2 μs spent for five lines of code. The current drawn by the Arduino was measured to be 42 mA. The total power dissipated was calculated as '5 Volts × 42 mA = 0.210 W' using the classical Power=$V \times I$ equation. Our finding is in agreement with the results of work [39], where power consumption of an Arduino UNO R3 board is measured as 0.150─0.394 W for different test conditions.

TABLE I. MEMORY SPACE CONSUMPTION

Used resource	Total available	Space used	Space remaining
Flash memory	32 KB	4.67 KB	27.33 KB
SRAM	2 KB	0.21 KB	1.79 KB

TABLE II. SECURITY PRIMITIVE TIME & POWER CONSUMPTION

Description	Time spent
Encryption time	282 μs
Decryption time	322 μs
Total IoT encryption-decryption latency	1492 μs
Encryption power consumption	0.210 W

The values in Table I and II indicate that our security additions have no significant impact on the performance of the IoT device, compared to the resources spent in wireless communications.

IV. SECURITY ANALYSIS

This section includes the informal and formal analyses of our proposed protocol. Formal analysis was carried out with well-accepted tools Scyther and AVISPA. But first of all, known attacks and our security measures will be explained, as our design uses Bluetooth for communication. Unfortunately, Bluetooth devices can be vulnerable to some attack types. For example, the attackers can capture the data using the Bluesnarfing attack on Bluetooth devices [35]. Fortunately, our proposed design entails that even if the transmitted messages are captured, it will not be possible to access the data inside, as it is encrypted.

A. Informal Security Analysis of Our Protocol Against Known Attacks

There are numerous types of attacks on IoT devices with limited resources. The proposed design secures reliable data communication between IoT devices against cyber-attacks, as explained below:

Replay Attack: The capture and replay of transmitted data are known as replay attacks [27]. Random numbers and pseudo names are standard tools for resisting replay attacks, and our designed protocol includes elements that update every session. Therefore, replay attacks cannot deceive our protocol.

Full-disclosure Attack: This type of attack reveals secrets of the communicating devices [40]. After revealing the secrets, all following information exchange is captured. AES encryption algorithm is known to protect message content. Our present study uses AES encryption and new AES session keys, each session. Therefore, our design is safe against disclosure attacks.

ID-theft Attack: ID-theft attack occurs when an unauthorized person obtains the device's ID and tracks the following sessions of the device [41]. Our proposed protocol uses changing pseudo-ID in each session. Additionally, if the old pSid1 of a sensor is used repeatedly to start a session, the gateway detects the attacking device using its counter.

Man-in-the-middle Attack: This type of attack is achieved by injecting an attacker between two communicating partners after changing transmitted messages on the fly [42]. In our study, the transmitted data is encrypted with AES, therefore injecting acceptable fake messages in the partners' exchange is impossible. Thus, our design is secure against a man-in-the-middle attack.

De-synchronization Attack: Selective interruption of packets can cause synchronization problems between devices and gateways. The counter in the gateway prevents synchronization attacks. The counter allows three starts from a sensor. If the sensor does not receive the new NP_1 in the 4^{th} step, the sensor still has three chances to complete around with the old $pSid_1$. The gateway blocks the unsuccessful sensor at the end of three attempts. Also, even if the gateway does not receive packets in the last step, the sensor can accept the packet is sent. Thus, the gateway and legitimate sensors remain synchronized while the attackers are blocked.

Identity Tracing Attack: In each session, identity tracking and ID capture are resisted by changing the pseudo sensor names. As the ID of the sensor changes in each session, it is impossible to communicate with the IoT gateway by reusing a captured ID.

Impersonation Attack: The IDs of the sensors are sent as cleartext which changes in each session. The attacker may want to guess the new ID by collecting the IDs of many sensors. However, the ID of each sensor is created as a pseudo-random number; therefore, our sensors cannot be impersonated.

DoS Attacks against the Gateway: DoS attacks aim at keeping IoT gateways busy [43]. Each sensor has a special counter in the IoT gateway to prevent the attack. Initially, the counter value is zero. If the communication is interrupted in the fourth or fifth steps in our protocol and the sensor fails to get a new pseudo-ID from the gateway, it will use the same ID in the next session. The counter value is incremented every time communication is attempted with the same pseudo-ID. After three attempts, the gateway becomes aware of the attack and disables the attacking ID. Thus, the IoT gateway is prevented from being constantly busy.

Clone Attack: The old cleartext ID is the mere information stolen from the sensor for cloning purposes.

However, since the gateway renews the ID in each session, the previous captured ID will be blocked after three communication attempts. So, clone attacks fail against our protocol.

B. Security Analysis of the Proposed Protocol Using Formal Analysis Tools Scyther and AVISPA

In recent years, the execution of automated authentication protocol validation tools has become popular. Our proposed protocol (Figure 4) was analyzed by not only one but two formal tools Scyther and AVISPA. Scyther and AVISPA are the most widely used tools for the formal analysis of security protocols [12], [16], [27], [30]. The analyses and verification results of both tools about our proposed protocol are explained in detail in the following two sub-sections.

1) Analysis Using Scyther Tool

Scyther is a popular tool for automatic verification of authentication protocols [44]. The tool has been utilized in many works for conducting formal analysis on the designed authentication protocols [45-47]. The tool allows checking for protocol design errors and behavior against attacks. In our work, Scyther version v1.1.3 for Windows was used. Our protocol was coded by the security protocol description language (SPDL), where the wearable IoT devices were given the 'sensor' role. The gateway was given the 'gateway' role. The five-step data transfers of the protocol were coded as "send" and "receive "parameters of Scyther. Thus, the data is exchanged by the "send" and "receive" parameters between role players. The "claim" events are used to control the security of the data transferred. The "claim" event requires parameters. These parameters are 'secret', 'alive', 'weakagree', session-key reveal ('SKR'), 'niagree' and 'nisynch'.

The 'secret' parameter indicates that the values of Rnd_1, Rnd_0, sensordata, NK_1, and NP_1 (protocol parameters of Figure 4) are expected to be secure during transmission. The first part in Figure 5 shows the above definition. 'alive' parameter indicates that communicating partners are expected to be alive. If the aliveness of the parties is verified, then the availability security feature of the protocol is ensured. Verifying the 'weakagree' parameter ensures that the protocol is immune to impersonation attacks [27]. The 'SKR' claim tests if the session key secrecy has been maintained. The use of 'niagree' parameter on both the 'sensor' and 'gateway' roles ensures that non-injective agreement (no messages can be injected into the exchange between the role players) is achieved. Therefore, data will be transmitted without corruption between the protocol parties. Finally, non-injective synchronization, i.e., the synchronization of five steps, is tested by the 'nisynch' parameter. Verification of this parameter ensures that all received messages have been sent by reciprocal and that the recipient received all sent messages.

The 'OK' declarations by the Scyther compiler in Figure 5 prove that the protocol was completed successfully, and no attacks were detected against the proposed protocol.

2) Analysis with AVISPA Tool

AVISPA is another popular tool used to verify authentication protocols in the literature. Currently, the AVISPA tool has a security protocol animator (SPAN). SPAN offers a graphical interface for interactive use of AVISPA features. In AVISPA, the tested authentication protocols are implemented using the high-level protocol specification language (HLPSL). The tool has four back-ends: On-the-fly model checker (OFMC), constraint-logic-based attack searcher (CL-AtSe), boolean satisfiability (SAT)-based model checker (SATMC), and tree automata-based protocol analyzer (TA4SP). OFMC and CL-AtSe checkers are more popular than SATMC and TA4SP.

AVISPA checks the submitted protocol and warns when it finds an attack against the protocol. Our protocol's HLPSL implementation was based on two roles: IoTsensor and IoTgateway. The analysis results of our protocol using OFMC are shown in Figure 6. The result on line 4 shows that our proposed mutual authentication protocol is 'SAFE' against attacks tested by AVISPA.

Scyther results : verify

Claim				Status		Comments
Hakhil	Sensor	Hakhil,Sensor1	Secret Rnd1	Ok	Verified	No attacks.
		Hakhil,Sensor2	Secret Rnd0	Ok	Verified	No attacks.
		Hakhil,Sensor3	Secret Sensordata	Ok	Verified	No attacks.
		Hakhil,Sensor4	Secret Nk1	Ok	Verified	No attacks.
		Hakhil,Sensor5	Secret Np1	Ok	Verified	No attacks.
		Hakhil,Sensor6	Alive	Ok	Verified	No attacks.
		Hakhil,Sensor7	Weakagree	Ok	Verified	No attacks.
		Hakhil,Sensor8	SKR k	Ok	Verified	No attacks.
		Hakhil,Sensor9	Niagree	Ok	Verified	No attacks.
		Hakhil,Sensor10	Nisynch	Ok	Verified	No attacks.
	Gateway	Hakhil,Gateway4	Secret Rnd1	Ok	Verified	No attacks.
		Hakhil,Gateway5	Secret Rnd0	Ok	Verified	No attacks.
		Hakhil,Gateway6	Secret Sensordata	Ok	Verified	No attacks.
		Hakhil,Gateway7	Secret Nk1	Ok	Verified	No attacks.
		Hakhil,Gateway8	Secret Np1	Ok	Verified	No attacks.
		Hakhil,Gateway9	Alive	Ok	Verified	No attacks.
		Hakhil,Gateway10	Weakagree	Ok	Verified	No attacks.
		Hakhil,Gateway11	SKR k	Ok	Verified	No attacks.
		Hakhil,Gateway12	Niagree	Ok	Verified	No attacks.
		Hakhil,Gateway13	Nisynch	Ok	Verified	No attacks.

Done.

Figure 5. Analysis result of the proposed protocol with the Scyther tool

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
    BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  C:\PROGRA~2\SPAN\testsuite\results\hakhil.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.01s
  searchTime: 0.00s
  visitedNodes: 9 nodes
  depth: 5 plies
```

Figure 6. Analysis result of the proposed protocol with AVISPA

Details of the protocol analysis are summarized after the 'SAFE' declaration of the tool. The tool confirms that our protocol had a bounded number of checker sessions. The location of the protocol HLPSL code file is also given in the results to ensure the user about the description analyzed. The output declares that the parsing of our protocol took 0.01 seconds. Searching for vulnerability took almost no system time, as our protocol is very concise. Our protocol is fully visited in only nine nodes. The depth of our protocol was 5, named as 'plies' (number of protocol steps).

V. Security Performance Comparison

This section presents the performance comparison of our proposed mutual authentication protocol with some previous protocols. Table III summarizes the security features of the comparison obtained from the related works. Protocols in works [18], [28] have not been included because any formal tools have not checked them.

The first compared feature is the number of verification tools used in the studies. Protocols of studies [27], [30] are proven by Scyther only, while the protocol in [16] was proved by AVISPA only. Our protocol is the only one proven by both Scyther and AVISPA. All of the protocols compared in Table III are mutual authentication protocols. The protocol of work [27] is completed in 4 steps, ours in 5, work [30] in 6, and work [16] in 18. It is evident that our design is very efficient in completing the authentication. Two protocols use symmetric encryption, while the other two fail to name their encryption method.

All of the protocols can resist the replay attack. While our protocol resists full-disclosure, ID theft, de-synchronization, identity tracking, and clone attacks, none of the rest of the protocols can resist these attack types. Work [16] and our protocol resist man-in-the-middle and impersonation attacks. Work [27] can resist only impersonation and replay attacks. Work [30] can resist man-in-the-middle attacks. However, only our design and work [30] can resist the DoS attack.

Table III. Comparison of Security Features Declared by the Authors

	Our	**[16]**	**[27]**	**[30]**
Formal verification tools used	Scyther & AVISPA	AVISPA	Scyther	Scyther
Mutual authentication	Yes	Yes	Yes	Yes
Number of scheme steps	5	18	4	6
Encryption method	Symmetric	Symmetric	Unspecified	Unspecified
Replay attack	Yes	Yes	Yes	Yes
Full-disclosure attack	Yes	No	No	No
ID-theft attack	Yes	No	No	No
Man-in-the-middle attack	Yes	Yes	No	Yes
De-synchronization attack	Yes	No	No	No
Identity tracking attack	Yes	No	No	No
Impersonation attack	Yes	Yes	Yes	No
DoS attack	Yes	No	No	Yes
Clone attack	Yes	No	No	No

It is evident from the comparison that our protocol is better verified with two popular verification tools. At the same time, our protocol is resistant to all the attacks listed in Table III, while the others are not. Now, our wearable sensor's data transmission in work [34] is secured against many attack types.

VI. Conclusion

Considering the increased use of IoT devices in smart cities and the necessity to secure the data they exchange, strong authentications in such systems are well accepted. Moreover, the authentication process is expected not to affect the performance of the devices. Thus, security and efficiency are two indispensable properties demanded in mutual authentications of low-resource IoT devices. In the present study, we propose a two-factor security scheme for IoT devices. The proposed authentication protocol was demonstrated as an application on the vulnerable wearable device of our previous work [34]. The collected data is stored in encrypted form, protecting data from unauthorized physical tampers. As a second security factor, a strong mutual authentication is provided for transferring the data. The proposed protocol was double-verified by two formal verification tools, Scyther, and automated validation of Internet security protocols and applications (AVISPA). Our protocol is verified as safe against nine known attacks. Hence, our design turns unprotected wearable smart textiles into secure smart textiles. The provided security is significant support for transferring IoT data securely and efficiently in smart cities. Therefore, the design presented in this study is a new security scheme that can be used in similar IoT and smart textiles applications.

References

[1] S. Li, L. D. Xu, S. Zhao, "The Internet of things: a survey," Information Systems Frontiers, vol. 17, no. 2, pp. 243-259, 2015. doi:10.1007/s10796-014-9492-7

[2] M. H. Özcanhan, "A new peculiarity to intelligent doors: security through information sharing," Pamukkale University Journal of Engineering Sciences, vol. 23, no. 5, pp. 581-587, 2017. doi:10.5505/pajes.2016.48753

[3] M. Kamal, M. Atif, H. Mujahid, T. Shanableh, A. R. Al-Ali et al., "IoT based smart city bus stops," Future Internet, vol. 11, no. 11, 2019. doi:10.3390/fi11110227

[4] H. A. Khattak, H. Farman, B. Jan, I. U. Din, "Toward integrating vehicular clouds with IoT for smart city services," IEEE Network, vol. 33, no. 2, pp. 65-71, 2019. doi:10.1109/MNET.2019.1800236

[5] N. Mohamed, J. Al-Jaroodi, I. Jawhar, "Towards fault tolerant fog computing for IoT-based smart city applications," In 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), pp. 0752-0757, 2019. doi:10.1109/CCWC.2019.8666447

[6] L. Zhao, J. Wang, J. Liu, N. Kato, "Optimal edge resource allocation in IoT-based smart cities," IEEE Network, vol. 33, no. 2, pp. 30-35, 2019. doi:10.1109/MNET.2019.1800221

[7] J. V. Jacobs, L. J. Hettinger, Y. H. Huang, S. Jeffries, M. F. Lesch et al., "Employee acceptance of wearable technology in the workplace," Applied Ergonomics, vol. 78, pp. 148-156, 2019. doi:10.1016/j.apergo.2019.03.003

[8] G. Aroganam, N. Manivannan, D. Harrison, "Review on wearable technology sensors used in consumer sport applications," Sensors, vol. 19, no. 9, 2019. doi:10.3390/s19091983

[9] Z. A. Alizai, N. F. Tareen, I. Jadoon, "Improved IoT device authentication scheme using device capability and digital signatures," In 2018 International Conference on Applied and Engineering Mathematics (ICAEM), pp. 1-5, 2018. doi:10.1109/ICAEM.2018.8536261

[10] I. Ali, S. Sabir, Z. Ullah, "Internet of things security, device authentication and access control: a review," International Journal of

Computer Science and Information Security (IJCSIS), vol. 14, no. 8, pp. 456-466, 2016

[11] K. Fan, Q. Luo, K. Zhang, Y. Yang, "Cloud-based lightweight secure RFID mutual authentication protocol in IoT," Information Sciences, vol. 527, pp. 329-340, 2020. doi:10.1016/j.ins.2019.08.006

[12] M. Alshahrani, I. Traore, "Secure mutual authentication and automated access control for IoT smart home using cumulative keyed-hash chain," Journal of information security and applications, vol. 45, pp. 156-175, 2019. doi:10.1016/j.jisa.2019.02.003

[13] M. Adeli, N. Bagheri, "Cryptanalysis of two recently proposed PUF based authentication protocols for IoT: PHEMAP and Salted PHEMAP," IACR Cryptol. ePrint Arch, 2019

[14] N. Li, D. Liu, S. Nepal, "Lightweight mutual authentication for IoT and its applications," IEEE Transactions on Sustainable Computing, vol. 2, no. 4, pp. 359-370, 2017. doi:10.1109/TSUSC.2017.2716953

[15] M. Saadeh, A. Sleit, K. E. Sabri, W. Almobaideen, "Hierarchical architecture and protocol for mobile object authentication in the context of IoT smart cities," Journal of Network and Computer Applications, vol. 121, pp. 1-19, 2018. doi:10.1016/j.jnca.2018.07.009

[16] B. Bera, A. K. Das, W. Balzano, C. M. Medaglia, "On the design of biometric-based user authentication protocol in smart city environment," Pattern Recognition Letters, vol. 138, pp. 439-446, 2020. doi:10.1016/j.patrec.2020.08.017

[17] D. Shah, V. Bharadi, "IoT based biometrics implementation on Raspberry Pi," Procedia Computer Science, vol. 79, pp. 328-336, 2016. doi:10.1016/j.procs.2016.03.043

[18] X. Sun, S. Men, C. Zhao, 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. doi:10.1002/sec.551

[19] Ö. Aydın, G. Dalkılıç, C. Kösemen, "A novel grouping proof authentication protocol for lightweight devices: GPAPXR+," Turkish Journal of Electrical Engineering & Computer Sciences, vol. 28, no. 5, pp. 3036-3051, 2020. doi:10.3906/elk-2004-5

[20] P. P. Gaikwad, J. P. Gabhane, S. S. Golait, "3-level secure Kerberos authentication for smart home systems using IoT," In 2015 1st International Conference on Next Generation Computing Technologies (NGCT), pp. 262-268, 2015. doi:10.1109/NGCT.2015.7375123

[21] A. W. Atamli, A. Martin, "Threat based security analysis for the Internet of things," International Workshop on Secure Internet of Things, pp. 35-43, 2014. doi:10.1109/SIoT.2014.10

[22] S. N. Firdous, Z. Baig, C. Valli, A. Ibrahim, "Modelling and evaluation of malicious attacks against the IoT MQTT protocol," In 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), pp. 748-755, 2017. doi:10.1109/iThings-GreenCom-CPSCom-SmartData.2017.115

[23] M. H. Özcanhan, S. Utku, "Attacking a PUF based mutual authentication protocol designed for Internet of things," International Journal of Computer Science and Software Engineering (IJCSSE), vol. 7, no. 11, pp. 270-274, 2018

[24] U. Çabuk, G. Kanakis, F. Dalkılıç, "LTE direct as a device-to-device network technology: use cases and security," Int. J. Adv. Res. Comput. Commun. Eng. (IJARCCE), vol. 5, no. 7, pp. 401-406, 2016. doi:10.17148/IJARCCE.2016.5779

[25] C. T. Chen, C. C. Lee, I. C. Lin, "Efficient and secure three-party mutual authentication key agreement protocol for WSNs in IoT environments," Plos one, vol. 15, no. 4, 2020. doi:10.1371/journal.pone.0232277

[26] K. N. Pallavi, V. R. Kumar, "Authentication-based access control and data exchanging mechanism of IoT devices in fog computing environment," Wireless Personal Communications, pp. 1-22, 2020. doi:10.1007/s11277-020-07834-w

[27] M. Nikooghadam, H. Amintoosi, "An improved secure authentication and key agreement scheme for healthcare applications," In 2020 25th International Computer Conference, Computer Society of Iran (CSICC), pp. 1-7, 2020. doi:10.1109/CSICC49403.2020.9050069

[28] C. C. Chang, H. D. Le, "A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks," IEEE Transactions on wireless communications, vol. 15, no. 1,pp. 357-366, 2015. doi:10.1109/TWC.2015.2473165

[29] A. K. Das, S. Kumari, V. Odelu, X. Li, F. Wu et al., "Provably secure user authentication and key agreement scheme for wireless sensor networks," Security and Communication Networks, vol. 9, no. 16, pp. 3670-3687, 2016. doi:10.1002/sec.1573

[30] A. Sangwan, V. R. Singh, "A secure authentication scheme for WiMax network and verification using scyther tool," International Journal of Applied Engineering Research, vol. 12, no. 11, 3002-3008, 2017

[31] P. K. Panda, S. Chattopadhyay, "A modified PKM environment for the security enhancement of IEEE 802.16e," Computer Standards & Interfaces, vol. 61, pp. 107-120, 2019. doi:10.1016/j.csi.2018.06.002

[32] A. K. Das, M. Wazid, N. Kumar, M. K. Khan, K. K. R. Choo et al., "Design of secure and lightweight authentication protocol for wearable devices environment," IEEE Journal of Biomedical and Health Informatics, vol. 22, no. 4, pp. 1310-1322, 2017. doi:10.1109/JBHI.2017.2753464

[33] V. Bianchi, M. Bassoli, G. Lombardo, P. Fornacciari, M. Mordonini et al., "IoT wearable sensor and deep learning: An integrated approach for personalized human activity recognition in a smart home environment," IEEE Internet of Things Journal, vol. 6, no. 5, pp. 8553-8562, 2019. doi:10.1109/JIOT.2019.2920283

[34] H. Dalkılıç, M. H. Özcanhan, H. Özdemir, "Wireless data transfer with the use of Internet of things (IoT) technologies in smart textiles," The Journal of The Textile Institute, Published Online: 17 June 2021, doi:10.1080/00405000.2021.1940035

[35] G. A. C. Montiel, F. B. Duque, L. A. P. Salazar, "BlueLock a tool to prevent Bluetooth attacks," Visión electrónica, vol. 14, no. 1, 2020. doi:10.14483/issn.2248-4728

[36] B. Mbarek, M. Ge, T. Pitner, "An efficient mutual authentication scheme for Internet of things," Internet of Things, vol. 9, 2020. doi:10.1016/j.iot.2020.100160

[37] D. Fang, Y. Qian, R. Q. Hu, "A flexible and efficient authentication and secure data transmission scheme for IoT applications," IEEE Internet of Things Journal, vol. 7, no. 4, pp. 3474-3484, 2020. doi:10.1109/JIOT.2020.2970974

[38] K. -H. Yeh, C. Su, K. -K. R. Choo, W. Chiu, "A novel certificateless signature scheme for smart objects in the Internet-of-things," Sensors, vol. 17, no. 5, 2017. doi:10.3390/s17051001

[39] M. Y. Al-Shorman, M. M. Al-Kofahi, O. M. Al-Kofahi, "A practical microwatt-meter for electrical energy measurement in programmable devices," Measurement and Control, vol. 51, no. 9-10, pp. 383-395, 2018. doi:10.1177/0020294018794350

[40] M. H. Özcanhan, G. Dalkılıç, S. Utku, "Cryptographically supported NFC tags in medication for better inpatient safety," Journal of medical systems, vol. 38, no. 8, 2014. doi:10.1007/s10916-014-0061-x

[41] T. Nandy, M. Y. I. B. Idris, R. M. Noor, M. L. M. Kiah, L. S. Lun et al., "Review on security of Internet of things authentication mechanism," IEEE Access, vol. 7, pp. 151054-151089, 2019. doi:10.1109/ACCESS.2019.2947723

[42] H. Wong, T. Luo, "Man-in-the-middle attacks on MQTT-based IoT using BERT based adversarial message generation," In KDD'20 Workshops: the 3rd International Workshop on Artificial Intelligence of Things (AIoT), 2020

[43] N. F. Syed, Z. Baig, A. Ibrahim, C. Valli, "Denial of service attack detection through machine learning for the IoT," Journal of Information and Telecommunication, pp. 1-22, 2020. doi:10.1080/24751839.2020.1767484

[44] E. Munivel, A. Kannammal, "New authentication scheme to secure against the phishing attack in the mobile cloud computing," Security and Communication Networks, 2019. doi:10.1155/2019/5141395

[45] N. E. Madhoun, G. Pujolle, "A secure cloud-based NFC payment architecture for small traders," In 2016 3rd Smart Cloud Networks & Systems (SCNS), pp. 1-6, 2016. doi:10.1109/SCNS.2016.7870562

[46] K. Yıldırım, G. Dalkılıç, N. Duru, "Security analysis of Hsiang m-coupon protocol," Journal of the Faculty of Engineering and Architecture of Gazi University, vol. 34, no. 4, pp. 1705-1724, 2019. doi:10.17341/gazimmfd.571490

[47] M. Eldefrawy, I. Butun, N. Pereira, M. Gidlund, "Formal security analysis of LoRaWAN," Computer Networks, vol. 148, pp. 328-339, 2019. doi:10.1016/j.comnet.2018.11.017