Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2020) 28: 3036 – 3051
© TÜBİTAK
doi:10.3906/elk-2004-5

A novel grouping proof authentication protocol for lightweight devices: GPAPXR+

Ömer AYDIN[1], Gökhan DALKILIÇ[2], Cem KÖSEMEN[3]*

[1]Department of Computer Engineering, The Graduate School of Natural and Applied Sciences, Dokuz Eylül University, İzmir, Turkey
[2]Department of Computer Engineering, Faculty of Engineering, Dokuz Eylül University, İzmir, Turkey
[3]Department of Computer Engineering, Faculty of Engineering, İzmir Bakırçay University, İzmir, Turkey

Received: 01.04.2020 • **Accepted/Published Online:** 19.06.2020 • **Final Version:** 25.09.2020

Abstract: Radio frequency identification (RFID) tags that meet EPC Gen2 standards are used in many fields such as supply chain operations. The number of the RFID tags, smart cards, wireless sensor nodes, and Internet of things devices is increasing day by day and the areas where they are used are expanding. These devices are very limited in terms of the resources they have. For this reason, many security mechanisms developed for existing computer systems cannot be used for these devices. In order to ensure secure communication, it is necessary to provide authentication process between these lightweight devices and the devices they communicate. The authentication process is the first step that allows the parties to trust each other for communication. Moreover, the authentication protocol should allow simultaneous verification of multiple lightweight devices. Therefore, grouping proof authentication protocol is required. In this study, a new grouping proof authentication protocol is developed for lightweight devices. The proposed protocol implemented on wireless identification and sensing platform passive RFID tag, uses embedded advanced encryption standard encryption method to encrypt transmitted data. Security of the protocol was first evaluated and verified theoretically, then by a tool used for automatic verification of security protocols, called Scyther tool.

Key words: Lightweight, wireless identification and sensing platform, grouping proof authentication protocol, security, Scyther tool, xorshift, xorshiftR+, GPAPXR+

1. Introduction

In recent years, emerging technologies, such as smart cards, radio frequency identification (RFID) tags, wireless sensor nodes, and the concept of Internet of things (IoT) has brought not only new solutions but also challenges in their scope of application. The proliferation of devices manipulating or transmitting sensitive and critical information requires more attention to security issues, because classical security algorithms and protocols cannot offer effective and feasible security solutions for these groups of devices. Thus, many lightweight cryptographic algorithms have been suggested in recent literature, including block ciphers [1–7] and hash functions [8–10]. The aim of lightweight security algorithms is to find a balanced solution for performance, speed, and security needs taking into account limitations such as storage and processing power. There are also lightweight protocols [11–13] as well as algorithms. One of the most important things for these protocols is random number generator. Xorshift pseudo-random number generator is a shift register generator, one of the well-known generators in the

*Correspondence: cem.kosemen@bakircay.edu.tr literature, and it was proposed by George Marsaglia [14]. This generator has been developed over time with modifications. There are many versions of that generator [15]. XorshiftR+ is a pseudorandom number generator developed over the xorshift class [16]. It was tested with NIST [17–21] and TestU01's [22] bigcrush tests and it passed all of them. It was implemented on wireless identification and sensing platform (WISP) passive RFID tag, and time and resource usage results are compared with those of some other versions of xorshift algorithms. It was shown to be suitable for lightweight devices.

An authentication protocol is a type of computer communication protocol. It is designed to prove the identity by transferring required information between the entities. This is the most important protection layer for secure communication within computer and IoT networks. The ease of identity theft in the virtual world has led to a search for specific identity authentication methods.

In RFID systems, the group authentication protocol provides a solution that allows for the authentication of multiple tagged objects. In the real world, the group authentication protocol has gained importance because tags are used on an increasingly wide range of objects.

In this paper, a new, effective, and lightweight grouping proof authentication protocol is proposed. This protocol uses the xorshiftR+ pseudorandom number generator that provides a lightweight solution to generate the random numbers required during communication between the RFID tags and the reader. It generates the random numbers using the algorithm called xorshiftR+ [16]. Therefore, we called our protocol as GPAPXR+ derived from the words "grouping proof authentication protocol using XorshiftR+". This protocol was implemented and tested on WISP passive RFID tag, and 256-bit advanced encryption standard (AES) was used to encrypt and decrypt the transported data. We used 256-bit AES on our test device "WISP" because it has built-in 256-bit AES chip, where 128-bit AES is secure for encrypting nonclassified data. The built-in AES feature allowed a faster, more scalable solution. We created a new scalable, lightweight, and secure grouping proof authentication protocol. This proposed protocol is structurally and practically tested on WISP passive RFID tag, and theoretically tested against the well-known attacks. Moreover, Scyther tool[1] was used to analyze the protocol [23]. According to the analysis and test results, our grouping proof authentication protocol resists the well-known attacks and is suitable for lightweight devices.

The rest of this article is organized as follows. Section 2 discusses previous and related work. In Section 3, details of material and methods used for the new grouping proof authentication protocol are given. Section 4 gives experimental results of performance, test, evaluation, and discussions. Finally, the paper is concluded in Section 5.

2. Related work

The literature contains many solutions for the authentication protocol, which is one of the main issues in the security of the lightweight devices. Studies are currently focusing on ways to overcome these challenges using innovative solutions. In this study, we examined previous pseudorandom number generator (PRNG) implementations and proposed grouping proof authentication protocol regarding the performance, statistical, and security properties.

Rostampour et al. [24] focused on the simultaneous identification of a set of objects tagged using RFID technology, an approach named RFID grouping proof authentication protocol. Using a method called authenticated encryption, they created a new authentication method that is scalable and secure. They argued

[1] Cremers C (2014). The Scyther Tool [online]. Website https://people.cispa.io/cas.cremers/scyther/. [accessed 28 May 2020]

that this protocol provides message integrity and confidentiality by not exceeding resource constraints for RFID systems. In the paper, they conducted a theoretical security analysis but do not use any tool such as the Scyther tool.

Çabuk et al. [16] proposed a new PRNG by modifying well-known xorshift algorithm called xorshiftR+, after developing many versions of the original xorshift128plus by changing parameters. Finally, three final versions were developed and compared. WISP passive RFID tag was used to implement these algorithms that were checked according to EPCGen2 standards, and the ENT (defined and detailed on www.fourmilab.ch/random), NIST, and TestU01 tests. The authors selected the best of the three versions based on the test results, resource usage and performance.

İbrahim and Dalkılıç [25] proposed a new mutual authentication protocol based on AES and elliptic curve cryptography (ECC) in the area of healthcare systems. The protocol was designed and implemented on WISP passive RFID tag. They used WISP passive RFID tag's built-in PRNG as the random number generator. However, the new mutual authentication protocol could not satisfy the timing limits of EPCGen2 standard because of the ECC needs, and also they gave no structural or statistical analysis of the PRNG.

Zhou et al. [26] proposed a secure and scalable grouping proof protocol with an encryption method for low-power and low-cost devices, functioning in two stages. In the first stage, it checks the group of the tag; in the second, it verifies the tag's ID. Encryption in this protocol is used for confidentiality and message integrity. The ECC is used as the encryption method. This protocol also uses offline approach.

Liew et al. [27] demonstrated a new authentication protocol using grouping proof to minimize the theft during transport of cargo with tagged items. This protocol provides integrity of all items in the cargo and transferability of the ownership of the tagged item. Theoretical security analysis of the proposed protocol was completed. The protocol was not tested on a real RFID system. Moreover, the protocol has not been examined in terms of time and resource usage.

Zhang et al. [28] proposed a scalable and lightweight grouping proof protocol for RFID tags, which confirms that a group of tags belong to the same group at the same time. This protocol uses exclusive-or (XOR) operations and pseudorandom number generation, which makes it suitable for large-scale grouping proof of lightweight devices. The reader first broadcasts a request message, and all tags around it respond with response messages. Based on the tag responses, the reader generates a grouping proof. An anticollision algorithm is used to identify the response messages. The authors used three oracle machines for security proof and did not test it on a real hardware scenario.

Zhou et al. [29] proposed an ECC-based offline grouping proof protocol. The reader is authorized to validate the tags without knowing their identities. It is claimed that this approach protects the privacy and the security of the tags, and defends against replay and impersonation attacks. In offline mode, no continuous connection between the reader and the verifier is needed, so the reader can proceed the grouping proof operation without the verifier. After proofing, the reader sends the data to the verifier, making the connection requirement more flexible. Due to the high resource requirement of ECC, this protocol cannot be used or is hard to run on lightweight and ultralightweight devices.

Tsai et al. [30] proposed a new grouping proof authentication protocol that enables partial ownership transfer by guaranteeing the integrity of the tagged cargoes. The protocol was analyzed against the security attacks theoretically. It is not tested with a tool such as the Scyther tool.

Luo and Yang [31] proposed a high-performance, secure ownership transfer protocol that provides the capability and security requirements to operate in RFID systems and existing ownership transfer environments.

They mentioned that their proposed protocol has a resistance to most of the known attacks. They only give information about confidentiality, replay, man-in-the-middle attacks, forward and backward secrecy, windowing problem, location privacy, asynchronous denial-of-service attack, and performance analysis. There is no evaluation about partial proof ability, physical disclosure, or cloning attack.

3. Material and methods

To develop an authentication protocol that is effective, secure, and lightweight, it is important to be very familiar with the characteristics of both the hardware and the environment. For the authentication protocol in the paper, a new grouping proof authentication protocol was developed using the WISP passive RFID tag with built-in 256 bit AES encryption, based on the use and modification of proven protocols.

3.1. XorshiftR+

XorshiftR+ is a lightweight and reliable pseudorandom number generator for lightweight devices used as IoT devices. It was developed by reducing and modifying the operations of the well-known PRNG xorshift+. Three reduced versions were created and they were implemented on WISP passive RFID tag. TestU01 test suite was used to measure their ability to generate random numbers. The performance of these three reduced versions were compared with some other lightweight random number generators. The best version was selected and called xorshiftR+. XorshiftR+ generator meets the three conditions [32,33] of EPCGen2 standards. It passes all tests of NIST test suite and TestU01's bigcrush test.

3.2. Grouping proof authentication protocol using XorshifR+: GPAPXR+

In this section, this new grouping proof authentication protocol is explained. Figure 1 shows the components of the RFID system implemented by the grouping proof authentication protocol developed within the scope of our article. Table 1 includes all the notations used in the protocol and Figure 2 shows the protocol. The protocol developed in this study consists of three phases detailed in subsections.

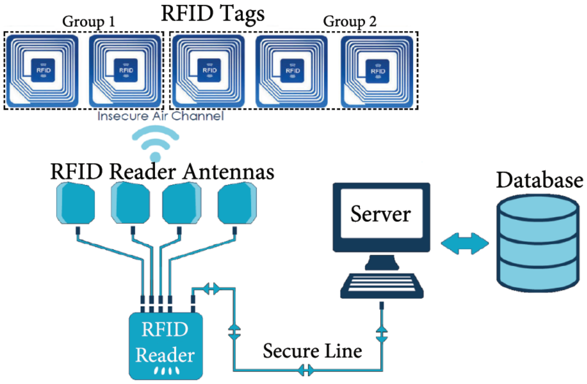


Figure 1. RFID system components with group authentication protocol applied [34].

RFID tags are located on the objects and the data communication between the RFID tags and the reader is on insecure air channel. The communication between the RFID reader and the server is assumed to be a secure connection. The database can be located in the same server or a separated server, and this communication line should also be secure.

Table 1. Notations.

k_g	A secret key for the group members
k_t	A secret key for the tag
ID_g	An identification code for the group
ID_t	An identification code for the tag
N_t	Nonce value generated by the tag
N_r	Nonce value generated by the RFID reader
MAC	Message authentication code
$\oplus$	Exclusive or operation
E(X, k)	X encrypted with key k using AES
D(E, k)	E decrypted with key k using AES
n	The number of the groups
m	The total number of the tags
i	Related tag number of a group
$M2^i$	Transmitted M2
${ID_g}^i$	Calculated ID_g(After decryption)
$M4^i$	Calculated M4 (After decryption)
$M3^i$	Calculated M3

3.2.1. Registration phase

At this stage, the server initiates the reader, the tags, and the groups with necessary records. Since the proposed protocol is based on the grouping method, there is a record for each group in the server's database. Two encryption keys, group key (k_g), and tag key (k_t) are set in each RFID tag. During the registration phase, the system behaves as follows: The server database contains a record for each tag. This entry contains ID_g, ID_t, k_g, k_t, N_t, and E (ID_t, k_t). In addition, each RFID tag stores ID_t, k_t, ID_g, and k_g. Figure 2 shows the functioning of the grouping proof authentication protocol between an RFID tag, a reader, and a server that includes a database. This scenario shows steps for one specific group (group ID is "i").

3.2.2. Authentication phase

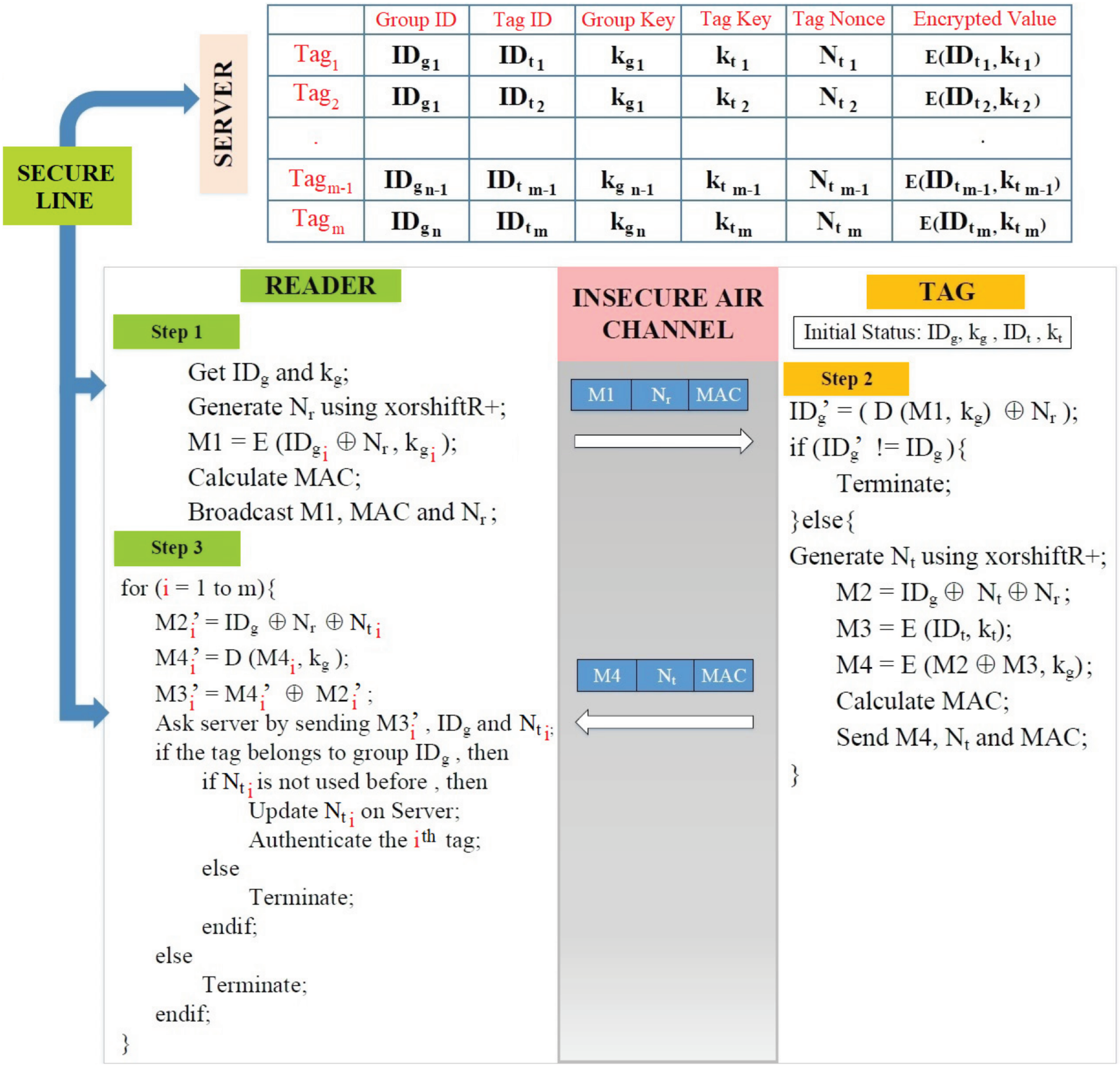
In this phase, the tags of the desired group were authenticated by running the following protocol steps below.

Step 1

- The server sends the required information to the RFID reader for the requested group.
- The reader runs xorshiftR+ algorithm and generates 4 random numbers (4 x 64 = 256 bit) and concatenates them to form the nonce value. This nonce value N_r is added to the message package to be sent to the RFID reader.
- The ID of the desired group and N_r are exposed to XOR operation and then the result is encrypted with k_g (desired group key) by using AES-256 and M1 is generated as shown in Eq. (1).

$$M1 = E\ (ID_g \oplus N_r,\ k_g) \qquad (1)$$

- M1 and N_r are concatenated, and then MAC is calculated.



	Group ID	Tag ID	Group Key	Tag Key	Tag Nonce	Encrypted Value
Tag_1	ID_{g1}	ID_{t1}	k_{g1}	$k_{t\,1}$	$N_{t\,1}$	$E(ID_{t\,1},k_{t\,1})$
Tag_2	ID_{g1}	ID_{t2}	k_{g1}	$k_{t\,2}$	$N_{t\,2}$	$E(ID_{t\,2},k_{t\,2})$
.						.
Tag_{m-1}	$ID_{g\,n-1}$	$ID_{t\,m-1}$	$k_{g\,n-1}$	$k_{t\,m-1}$	$N_{t\,m-1}$	$E(ID_{t\,m-1},k_{t\,m-1})$
Tag_m	$ID_{g\,n}$	$ID_{t\,m}$	$k_{g\,n}$	$k_{t\,m}$	$N_{t\,m}$	$E(ID_{t\,m},k_{t\,m})$

Step 1

```
Get $ID_g$ and $k_g$;
Generate $N_r$ using xorshiftR+;
$M1 = E(ID_{g_i} \oplus N_r, k_{g_i})$;
Calculate MAC;
Broadcast M1, MAC and $N_r$;
```

Step 3

```
for (i = 1 to m){
    $M2_i' = ID_g \oplus N_r \oplus N_{t_i}$
    $M4_i' = D(M4_i, k_g)$;
    $M3_i' = M4_i' \oplus M2_i'$;
    Ask server by sending $M3_i'$, $ID_g$ and $N_{t_i}$;
    if the tag belongs to group $ID_g$, then
        if $N_{t_i}$ is not used before , then
            Update $N_{t_i}$ on Server;
            Authenticate the i^th tag;
        else
            Terminate;
        endif;
    else
        Terminate;
    endif;
}
```

Step 2

```
$ID_g' = (D(M1, k_g) \oplus N_r)$;
if ($ID_g'$ != $ID_g$){
    Terminate;
}else{
Generate $N_t$ using xorshiftR+;
    $M2 = ID_g \oplus N_t \oplus N_r$;
    $M3 = E(ID_t, k_t)$;
    $M4 = E(M2 \oplus M3, k_g)$;
    Calculate MAC;
    Send M4, $N_t$ and MAC;
}
```

Figure 2. GPAPXR+ steps.

- Reader concatenates M1, N_r, and MAC, and then broadcasts this information to all tags in its active field.

Step 2

After receiving the encrypted message, each tag decrypts message M1 using k_g, the result and transmitted N_r are exposed to XOR operation and the tag finds the calculated group ID from the message, called ID_g'. It compares the group ID with the previously stored group ID on its own, and a match shows that the tag is a member of the desired group. Otherwise, it is verified that the tag does not belong to the group being inquired by the reader, and the communication is interrupted at this point. If the group ID matches, the tag generates random numbers using the algorithm xorshiftR+ as the nonce value (N_t). On the tag side, if it is needed, zeros are appended to the random number for the missing bits and then XOR operation is proceeded by matching three numbers bit by bit. M2 and M3 are as in Eqs. (2) and (3), respectively:

$$\mathrm{M2} = \mathrm{ID}_g \oplus \mathrm{N}_t \oplus \mathrm{N}_r \tag{2}$$

$$\mathrm{M3} = \mathrm{E}(\mathrm{ID}_t,\ \mathrm{k}_t) \tag{3}$$

M2 and M3 are encrypted with (k_g) after being exposed to XOR operation.

$$\mathrm{M4} = \mathrm{E}\ (\mathrm{M2} \oplus \mathrm{M3},\ \mathrm{k}_g) \tag{4}$$

The tag calculates MAC from M4 and N_t, and then MAC, N_t, and M4 are concatenated and sent back to the reader as the response message.

Step 3

The reader always waits for tag responses. If the response is received, the reader calculates some values in a loop. In Figure 2, m indicates the number of tags in each group. On the other hand, n indicates the total number of groups. The reader calculates M2', M4', and M3', respectively as in Eqs. (5–7) for each tag response separately.

$$\mathrm{M2'} = \mathrm{ID}_g \oplus \mathrm{N}_r \oplus \mathrm{N}_t \tag{5}$$

$$\mathrm{M4'} = \mathrm{D}\ (\mathrm{M4},\ \mathrm{k}_g); \tag{6}$$

$$\mathrm{M3'} = \mathrm{M4'} \oplus \mathrm{M2'}; \tag{7}$$

Then it sends M3' along with ID_g, and N_t to the server and waits for a response from the server.

3.2.3. Verification phase

The server retrieves M3' sent from the reader and searches the database. It compares M3' with E (ID_t, k_t) encrypted value on the database and if it finds a match, it compares the N_t value sent by the reader with the N_t value in the corresponding row in the database. If there is a match in N_t, the server returns a replay attack warning, reporting that the nonce value was previously used. When M3 values are matched and if the new nonce value was not used before, the existing N_t value in the database is updated with the new value from the reader, and the values such as ID_t and k_t of the corresponding record are returned to the reader.

4. Security analysis of the GPAPXR+

This section contains an analysis of the security level based on various methods of the GPAPXR+. GPAPXR+ was evaluated against some well-known RFID attacks to discover whether or not it is strong against them. In addition, the protocol was simulated with a special programming language named "the Scyther tool input language" and evaluated using the Scyther protocol security analysis tool developed by Cremers[1]. Then, it was examined in terms of providing security services and compared with known authentication protocols.

Firstly, it is assumed that the attacker has the following abilities, access, information, and resources [35]:

1. The attacker can listen to all messages between the tags and the reader.
2. The attacker may block the transmitted data on the communication channel.

3. The attacker may send a message to the other party as a tag or a reader.

4. The attacker has no access to hidden parameters, but can access all functions or operations, such as PRNG, encryption, and XOR.

5. The attacker can process all transmitted messages, such as creating, modifying, and deleting them and returning them to the communication channel.

4.1. Theoretical security analysis

The new protocol has been examined theoretically against known security attacks. During this investigation, the protocol's resistance to traceability, replay, physical disclosure, impersonation, desynchronization, denial of service, man in the middle, cloning, and eavesdropping attacks were questioned.

4.1.1. Traceability attack

The traceability attack can be performed if the tag sends fixed responses to the reader. In GPAPXR+, to prevent this type of attack, the tag adds the nonce value (N_t) to each message. N_t is produced by a PRNG "xorshiftR+" that passes NIST STS and TestU01 tests, as mentioned in the previous section. As well as the addition of the N_t value at the end of messages, the encrypted text and MAC value are also changed in each message since nonce value is also added to the messages before encryption process. Therefore, the proposed protocol includes defense against traceability attack. However, it is possible to find out the group to which the tag belongs. In this case, any attacker who stores and sends a valid M1, nonce, and MAC value will receive a response from any tag in the group, although it cannot associate responses with a specific tag. On the other hand, the attacker will need to have k_g and ID_g values to calculate a valid M1 and MAC value. Moreover, the attacker has to guess the used encryption algorithm.

4.1.2. Replay attack

It is assumed that an attacker could process or record messages transmitted during the successful authentication process in this type of attack. At the same time, when an RFID reader issues a request, it is assumed that the attacker can participate in the process instead of a valid tag. The attacker can attempt to intervene in this process by resending the recorded information, but cannot recreate a valid M4 value. This is because M4 value is encrypted, and N_t value in M2, one of the values that make up M4, is continuously changing. Previous N_t values were saved on the server database so this attack will not be successful.

4.1.3. Physical disclosure attack

In this type of attack, it is assumed that the attacker can physically gain access to a tag that is a member of the specified group, and the tag's all hidden information. If an attacker can execute attacks such as traceability, identity impersonation, or message repetition attacks, this attack is known as a physical disclosure attack. In our proposed grouping proof authentication protocol, if the attacker is able to access the tag physically, it can seize the hidden parameters of a tag, such as ID_g, k_g, ID_t, and k_t. Although M4 sent by other tags will be decrypted using k_g, the attacker will not be able to acquire the ID_t information of other tags in the group, because each tag encrypts its ID_t with its own unique k_t. Thus, the attacker can produce M4 for a fake tag with k_g but it is not possible to authenticate this tag. In verification phase, $M3^i$ is searched in the database. This value is an encrypted text stored in database. Attacker cannot create a matching encrypted text for a fake tag, so the proposed protocol is strong against physical disclosure attacks. It is not possible to threaten protocol security by seizing hidden parameters of a tag.

4.1.4. Impersonation attack

In this type of attack, the attacker convinces the RFID reader that the received messages were sent by a valid tag. On the tag's side, the values of M3 and M4 are encrypted using AES algorithm. While the attacker needs ID_t and k_t values to calculate and encrypt M3 value, to calculate M4 value it needs k_g value in addition to M3. Also, it cannot estimate the MAC value because of not having k_t value; therefore, the tag cannot impersonate the identity, and the protocol can resist this attack.

4.1.5. Desynchronization attack

Desynchronization attack can be done by ensuring that the sequence number in incoming packets differs from the expected sequence number. Especially in cases where the parties use counters, mutual synchronization is disrupted. Both sides do not have the same counter value. In the case of asynchrony, both communication endpoints discard received packets. At this point, attackers can infiltrate the system and provide packages with the correct sequence number. Attackers can even change or redirect communication.

In GPAPXR+ scheme, M4 and N_t nonce values are taken by the reader and $M4^i$ is created using N_t and the values stored on the reader side. N_t value is also used to create M4. M4 value is encrypted data and cannot be changed by the attackers. The invariance of M4 and other values is guaranteed by MAC. On the other hand, there is not a counter to synchronize neither reader side nor tag side. Therefore, we can say that the proposed scheme can resist desynchronization attacks.

4.1.6. Denial of service attack

Denial of service attack is an attack that the services of a computer, machine, or network device connected to the Internet are blocked temporarily or permanently. The proposed protocol is not susceptible to the denial-of-service attacks because it has resistance to desynchronization attacks. Reader and the tag are available to accept communication all the time.

4.1.7. Man in the middle attack

Transferred data between the reader and the tag is encrypted by AES-256. The attacker has to break AES-256 encryption to apply man-in-the-middle attack. This is not possible.

4.1.8. Cloning attack

Cloning attack is not applicable for the proposed scheme because obtaining the tag ID is computationally infeasible under the AES-256.

4.1.9. Eavesdropping

Attacker can listen to the communication between the tag and the reader but sent critical data is encrypted, so attacker cannot estimate the plain text of that critical data.

4.2. Security analysis with the Scyther tool

The Scyther security analysis tool provides an interface to the user using the Python code infrastructure. This interface allows the intended users to perform security analysis of the protocol and to understand the results without difficulty. GPAPXR+ was tested using the Scyther tool. The Scyther tool has the ability to try multiple attacks to the authentication protocols and shows the results of these attacks. Our Scyther tool model examines three claims, namely Secret, Nisynch, and Niagree. These claims can be defined as follows: Niagree is noninjective synchronization and ensures there is no interference in the content of the message exchanged between the receiver and the sender and the communication is completed in the protocol. Nisynch guarantees that communication packets are exchanged in the expected order, and also the run of the protocol is Niagree [36]. "Secret claims" guarantee that the messages between the sender and the receiver are secret [37]. As a result of the examination, it was determined that the protocol passed all the attack tests, as shown in the Scyther tool result screen in Figure 3[2]. The Scyther tool results show that it is impossible to attack the authentication operations in each layer; therefore, the authentication protocol is considered secure according to the Scyther tool results.

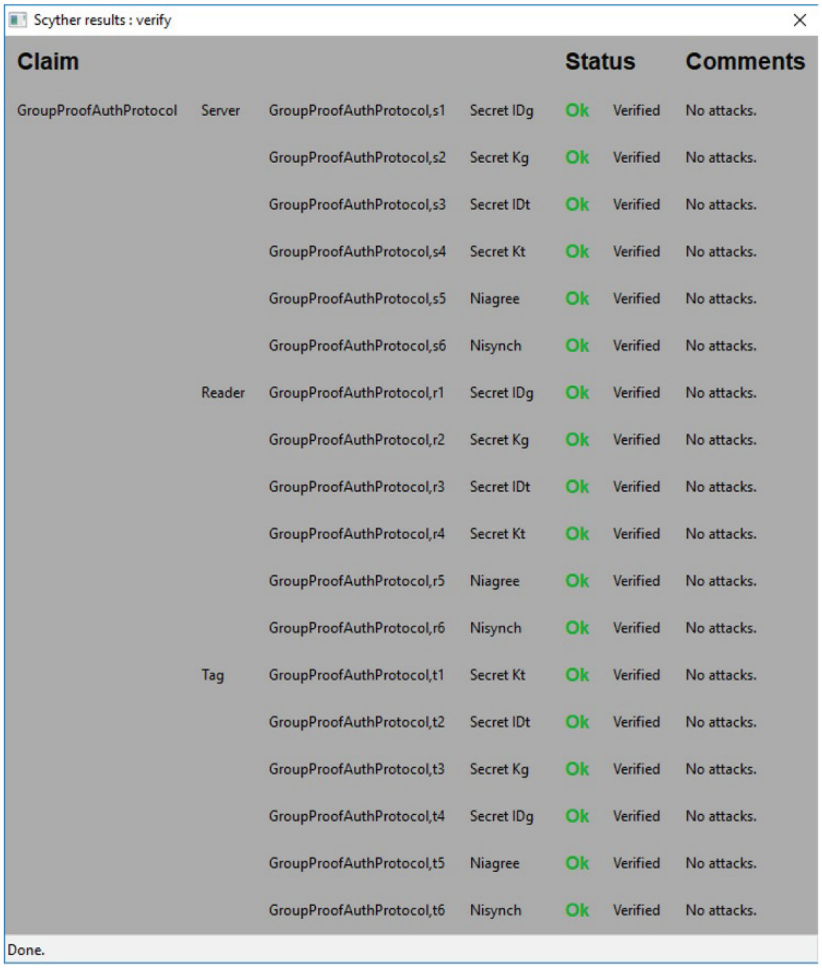
4.3. Security service analysis

It is possible that within the interaction area of the reader, some tags do not belong to the corresponding group; therefore, it is important to prevent the authentication protocol from including valid but unwanted tags in the authentication process. First of all, our grouping proof protocol checks the membership of the tag. The reader calculates $M2^i = ID_g \oplus N_t \oplus N_r$ and decrypts M4 value and calculates $M4^i$ value. It calculates $M3^i$ as $M3^i = M4^i \oplus M2^i$. Finally, $M3^i$ is sent to the server, which searches for it in the database. If it does not belong to the desired group, the server sends a reject message. This is the strategy in the proposed protocol to remove valid but unwanted tags.

The new protocol has the ability to verify the identity of a number of tags in a group or subgroups, called partial-proof. Partial-proof is performed on the server side. The server transfers the requested group information to the RFID reader that sends the responses of the tags to the server, and runs the authentication process. The server compares each tag in the desired group with the value of $E(ID_t, k_t)$ stored in the database. Partial-proof is valid if the server can find the record of the desired group. Also, if one or more tags are separated from the group, the server can run the authentication process based on the remaining tags. Thus, the proposed protocol has the ability to perform the authentication for groups of one or more tags, regardless of the number of tags in the groups.

The reader sends its encrypted group ID to the tag. Tag decrypts the sent ID and compares it with the stored group ID. If they are equal, the tag authenticates the reader. On the other hand, the tag sends an encrypted message to the reader. The reader decrypts the sent data and asks the server, if the server says that everything is okay, the reader authenticates the tag and the mutual authentication process is completed.

[2] DEU CENG SRG (2020). The Scyther tool protocol modeling files [online]. Website http://srg.cs.deu.edu.tr/publications/2020/gpapxr/. [accessed 28 May 2020]



Claim				Status		Comments
GroupProofAuthProtocol	Server	GroupProofAuthProtocol,s1	Secret IDg	Ok	Verified	No attacks.
		GroupProofAuthProtocol,s2	Secret Kg	Ok	Verified	No attacks.
		GroupProofAuthProtocol,s3	Secret IDt	Ok	Verified	No attacks.
		GroupProofAuthProtocol,s4	Secret Kt	Ok	Verified	No attacks.
		GroupProofAuthProtocol,s5	Niagree	Ok	Verified	No attacks.
		GroupProofAuthProtocol,s6	Nisynch	Ok	Verified	No attacks.
	Reader	GroupProofAuthProtocol,r1	Secret IDg	Ok	Verified	No attacks.
		GroupProofAuthProtocol,r2	Secret Kg	Ok	Verified	No attacks.
		GroupProofAuthProtocol,r3	Secret IDt	Ok	Verified	No attacks.
		GroupProofAuthProtocol,r4	Secret Kt	Ok	Verified	No attacks.
		GroupProofAuthProtocol,r5	Niagree	Ok	Verified	No attacks.
		GroupProofAuthProtocol,r6	Nisynch	Ok	Verified	No attacks.
	Tag	GroupProofAuthProtocol,t1	Secret Kt	Ok	Verified	No attacks.
		GroupProofAuthProtocol,t2	Secret IDt	Ok	Verified	No attacks.
		GroupProofAuthProtocol,t3	Secret Kg	Ok	Verified	No attacks.
		GroupProofAuthProtocol,t4	Secret IDg	Ok	Verified	No attacks.
		GroupProofAuthProtocol,t5	Niagree	Ok	Verified	No attacks.
		GroupProofAuthProtocol,t6	Nisynch	Ok	Verified	No attacks.

Done.

Figure 3. The Scyther tool results.

The proposed protocol resists the desynchronization and denial-of-service (DoS) attacks, so we can say that the availability is maintained.

The tag ID and all critical data are sent to the reader encrypted, so it is confidential as breaking the AES-256 encryption is computationally infeasible. We can say that confidentiality of tag anonymity, traceability, location, and information privacy are maintained by that reason. The previous/current confidential information cannot be compromised by the attacker because all critical data including the tag ID are transmitted as encrypted text. Forward and backward security are maintained.

The transfer of ownership means that the new owner's server has inherited the tag authorization; therefore, the identity and the authorization must be transferred securely. In our scenario, authorization and other critical information are stored in the server's database. If this critical data is transferred to any other server, it means that the ownership is transferred.

4.4. Comparison with the other authentication protocols

In our study, we identified comparison metrics as resisting security threats, providing security services, whether grouping proof is present or not, have a PRNG passing well-known test suites, encryption function used, verification method, whether implemented on a real device or not, and EPC compatibility. We created a comparison table using these metrics. The protocols satisfying those metrics are detailed in Table 2. Information about the protocols in reference studies shown in Table 2 is given under the section “Related work”. Security threats and services are also evaluated in the table. Moreover, some well-known authentication protocols are given in that table to compare with GPAPXR+. We can see that GPAPXR+ has resistance against traceability&tracking, replay, physical disclosure, impersonation, desynchronization, DoS, man-in-the-middle, cloning, and eavesdropping attacks. Moreover, it supports mutual authentication, confidentiality, availability, forward&backward security, ownership transferability, tag anonymity, and location&information privacy services.

GPAPXR+ uses AES to encrypt and decrypt the transmitted data. The protocol (GPAPXR+) was tested and realized on WISP passive RFID tag and complies with the EPC Gen2 standards. Other protocols provide various security services and resist attacks but GPAPXR+ stands out among them. GPAPXR+ was implemented and tested on a real device. The other protocols given in Table 2 are mostly verified manually or theoretically. GPAPXR+ uses a PRNG that passes NIST and TestU01 tests while none of the previous studies in Table 2 has an evidence of using PRNG passing some strict tests. Some of the previous studies are not compatible with EPC standards. Some protocols in the table are not suitable for grouping proof. Most of the protocols in the table are not verified by a tool and they are verified theoretically. On the other hand, GPAPXR+ is verified by Scyther tool and also it is verified theoretically. GPAPXR+ was implemented on WISP passive RFID tag, but most of the other protocols were not implemented on a real device.

GPAPXR+ can use AES-128 or AES-256, so this is an important advantage in security. In GPAPXR+, AES-256 was used because WISP passive RFID tags has built-in AES-256. For other devices, alternative secure encryption algorithms can be used to encrypt the transmitted data. We can say that our protocol can use any cryptographic algorithm to secure transmitted data and it already uses an encryption method such as AES. For this reason, it is not recommended to be used in RFID tags that do not have the capacity to make secure encryption.

There are also some disadvantages of the proposed protocol. GPAPXR+ protocol runs on fully fledged devices. Therefore, it does not work on IoT devices that are not fully fledged. AES is used for encryption processes. Other encryption algorithms can be used instead of this algorithm, but they will not work on devices that cannot run an encryption algorithm. To generate a random number, they will need to produce seeds. Therefore, they need a hardware entropy source to produce seeds.

5. Conclusion

This study presents a proposal for a solution on existing and pending security challenges on lightweight devices in consideration of resource and time constraints. New solution is offered for authentication that is the important point of security for lightweight devices. The WISP passive RFID tag was used as the lightweight device to conduct tests and experiments. WISP passive RFID tag has built-in sensors, 256-bit AES encryption, and can be programmed and therefore was selected for its high usability and applicability in the scope of future technologies.

Previous authentication protocols were examined in the scope of this study, and group proof authentication protocols, capable of obtaining effective and flexible solutions under the desired resource limitations were considered. This resulted in the development of the new group proof authentication protocol discussed in this article. A secure solution is produced using WISP passive RFID tag’s built-in 256 bit AES encryption with this new protocol and xorshifR+ is used for random number generation required for nonce values on both RFID

Table 2. Comparison of authentication protocols.

Procotol	Theoretical security analysis									Security services analysis							Other features						
	Traceability/tracking attack	Replay attack	Physical disclosure attack	Impersonation attack	Desynchronization	DoS	Man-in-the-middle	Cloning attack	Eavesdropping	Mutual authentication	Confidentiality	Availability	Forward/backward security	Ownership transferablity	Tag anonymity	Location/information privacy	Grouping proof	Using a secure PRNG	Function based on	Verificated by a tool	Verified theoretically	Implemented on a real device	EPC compatibility
GPAPXR+	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	AES	✓	✓	✓	✓
[24]	✓	✓	✓	✓	✓	?	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	?	AE	X	✓	X	X
[25]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	?	✓	✓	X	X	ECC&AES	X	✓	✓	✓
[26]	?	✓	?	✓	?	?	?	?	?	?	?	?	?	?	✓	✓	✓	?	ECC	X	✓	X	?
[27]	?	✓	?	?	?	✓	?	?	?	?	?	?	✓	✓	?	?	✓	?	HASH	X	✓	X	?
[28]	?	✓	?	?	✓	?	?	?	?	✓	?	?	?	?	✓	?	✓	?	HASH&AES	X	✓	X	?
[29]	✓	✓	?	✓	?	?	?	?	✓	?	?	?	?	?	✓	✓	✓	?	ECC	X	✓	X	?
[30]	?	✓	?	?	?	✓	?	?	?	?	?	?	✓	✓	?	?	✓	?	PKC	X	✓	X	✓
[31]	✓	✓	?	?	?	✓	✓	?	✓	?	✓	?	✓	✓	?	✓	✓	?	AES&RSA	X	✓	X	?

✓:Condition is fulfilled, X: Not fullfilled, ?: No information in the article, AES: Advanced encryption standard
ECC: Elliptic-curve cryptography, AE: Authenticated encryption, PKC: Public key cryptography

reader and RFID tag. This article describes the protocol's structural design, phases, and process steps, and also provides the security analysis of the protocol based on an evaluation of its effectiveness against known attack methods. Also, it is evaluated in terms of provided security services. In addition, protocol analysis is performed by using the Scyther security analysis tool. It is observed that the protocol successfully prevents well-known forms of attack.

GPAPXR+ was designed to resist many security threats and to perform many security services. It has been implemented and tested on WISP passive RFID tag. Along with this, the working ability in different environments and devices can be monitored for different types of attacks that may be encountered in the future. It can also be run on different RFID tags, IoT devices, and can be tested with criteria such as time, security, and resource usage.

Acknowledgment

This study is supported by The Scientific & Technological Research Council of Turkey (TÜBİTAK) with the project number 215E225.

References

[1] Canniere CD, Dunkelman O, Knezevic M. KATAN and KTANTAN – a family of small and efficient hardware-oriented block ciphers. In: International Workshop on Cryptographic Hardware and Embedded Systems CHES 2009; Lausanne, Switzerland; 2009. pp. 272-288. doi: 10.1007/978-3-642-04138-9_20

[2] Knudsen L, Leander G, Poschmann A, Robshaw MJB. PRINTcipher: a block cipher for ic-printing. In: 12th International Workshop Cryptographic Hardware and Embedded Systems CHES 2010; Santa Barbara, USA; 2010. pp. 16-32. doi: 10.1007/978-3-642-15031-9_2

[3] Guo J, Peyrin T, Poschmann A, Robshaw M. The led block cipher. In: 13th International Workshop Cryptographic Hardware and Embedded Systems CHES 2011, Nara, Japan; 2011. pp. 326-341. doi: 10.1007/978-3-642-23951-9_22

[4] Gong Z, Nikova S, Law YW. A new family of lightweight block ciphers. In: Juels A, Paar C (editors) RFID. Security and Privacy. RFIDSec 2011. Lecture Notes in Computer Science, vol 7055. Berlin, Heidelberg: Springer, 2012. pp. 1-18. doi: 10.1007/978-3-642-25286-0_1

[5] Shibutani K, Isobe T, Hiwatari H, Mitsuda A, Akishita T et al. Piccolo:an ultra-lightweight blockcipher. In: International Workshop on Cryptographic Hardware and Embedded Systems CHES 2011: Cryptographic Hardware and Embedded Systems; Nara, Japan; 2011. pp. 342–357. doi: 10.1007/978-3-642-23951-9_23

[6] Wu W, Zhang L. LBlock: a lightweight block cipher. In: International Conference on Applied Cryptography and Network Security ACNS 2011: Applied Cryptography and Network Security; Nerja, Spain; 2011. pp. 327–344. doi: 10.1007/978-3-642-21554-4_19

[7] Beaulieu R, Shors D, Smith J, Treatman-Clark S, Weeks B et al. The SIMON and SPECK families of lightweight block ciphers. In: 2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC); San Francisco, CA, USA; 2015. doi: 10.1145/2744769.2747946

[8] Bogdanov A, Knezevic M, Leander G, Toz D, Varıcı K et al. SPON-GENT: a lightweight hash function. In: International Workshop on Cryptographic Hardware and Embedded Systems CHES 2011: Cryptographic Hardware and Embedded Systems; Nara, Japan; 2011. pp. 312-325. doi: 10.1007/978-3-642-23951-9_21

[9] Guo J, Peyrin T, Poschmann A. The PHOTON family of lightweight hash functions. In: Annual Cryptology Conference CRYPTO 2011: Advances in Cryptology; Santa Barbara, CA, USA; 2011. pp. 222–239. doi: 10.1007/978-3-642-22792-9_13

[10] Aumasson JP, Henzen L, Meier W, Naya-Plasencia M. Quark: a lightweight hash. Journal of Cryptology 2013; 26(2): 313-339. doi: 10.1007/s00145-012-9125-6

[11] Lee JY, Lin WC, Huang YH. A lightweight authentication protocol for internet of things. In: 2014 International Symposium on Next-Generation Electronics (ISNE); New York, USA; 2014. pp. 1-2.

[12] Özcanhan MH, Dalkılıç G. Mersenne twister-based RFID authentication protocol. Turkish Journal of Electrical Engineering & Computer Sciences 2015; 23: 231-254. doi: 10.3906/elk-1212-95

[13] Armknecht F, Hamann M, Mikhalev V. Lightweight authentication protocols on ultra-constrained RFIDs-myths and facts. In: International Workshop on Radio Frequency Identification: Security and Privacy Issues; 2015. pp. 1-18. Springer, Cham.

[14] Marsaglia G. Xorshift RNGs. Journal of Statistical Software 2003; 8(14): 1-6. doi: 10.18637/jss.v008.i14

[15] Vigna S. Further scramblings of Marsaglia's xorshift generators. Journal of Computational and Applied Mathematics 2017; 315: 175-181. doi: 10.1016/j.cam.2016.11.006

[16] Çabuk UC, Aydın Ö, Dalkılıç G. A random number generator for lightweight authentication protocols: xorshiftR+. Turkish Journal of Electrical Engineering & Computer Sciences 2017; 25(6): 4818-4828. doi: 10.3906/elk-1703-361

[17] Bassham LE. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. NIST Special Publication 800-22rev1a. Gaithersburg, MD, USA: NIST, 2010.

[18] Rukhin A, Soto J, Nechvatal J, Smid M, Barker E et al. A statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST Special Publication NIST SP 800-22 Rev 1a, Gaithersburg, MD, USA: NIST, 2010.

[19] Barker E, Kelsey J. Recommendation for Random Number Generation Using Deterministic Random Bit Generators. NIST Special Publication 800-90A. Gaithersburg, MD, USA: NIST, 2012.

[20] Turan MS, Barker E, Kelsey J, McKay KA, Baish ML et al. Recommendation for the Entropy Sources Used for Random Bit Generation. NIST Special Publication 800-90B Second Draft. Gaithersburg, MD, USA: NIST, 2016.

[21] Barker E, Kelsey J. Recommendation for Random Bit Generator (RBG) Constructions. NIST Special Publication 800-90C Second Draft. Gaithersburg, MD, USA: NIST, 2016.

[22] L'Ecuyer P, Simard R. TestU01: A Software Library in ANSI C for Empirical Testing of Random Number Generators User's Guide Document. Montreal, Canada: University of Montreal, 2014.

[23] Cremers C. The scyther tool: verification, falsification, and analysis of security protocols. In: International Conference on Computer Aided Verification CAV 2008: Computer Aided Verification; Princeton, NJ, USA; 2008. pp. 414-418. doi: 10.1007/978-3-540-70545-1_38

[24] Rostampour S, Bagheri N, Hosseinzadeh M, Khademzadeh A. An authenticated encryption based grouping proof protocol for RFID systems. Security and Communication Networks 2016; 9(18): 5581-5590. doi: 10.1002/sec.1718.

[25] İbrahim A, Dalkılıç G. An advanced encryption standard powered mutual authentication protocol based on elliptic curve cryptography for RFID, proven on WISP. Journal of Sensors 2017; 1: 1-10. Article ID 2367312. doi: 10.1155/2017/2367312

[26] Zhou Z, Liu P, Liu Q, Wang G. An ECC-based off-line anonymous grouping-proof protocol. In: Wang G, Atiquzzaman M, Yan Z, Choo KK (editors) Security, Privacy, and Anonymity in Computation, Communication, and Storage. SpaCCS. Lecture Notes in Computer Science 10656. Cham: Springer, 2017, pp. 186-200. doi: 10.1007/978-3-319-72389-1_16

[27] Liew WT, Tsai KY, Luo JN, Yang MH. Novel designated ownership transfer with grouping proof. In: 2017 IEEE Conference on Dependable and Secure Computing; Taipei, Taiwan; 2017. pp. 433-440. doi: 10.1109/DESEC.2017.8073863

[28] Zhang W, Qin S, Wang S, Wu L, Yi B. A new scalable lightweight grouping proof protocol for RFID systems. Wireless Personal Communications 2018; 103: 133-143. doi: 10.1007/s11277-018-5430-1

[29] Zhou Z, Liu P, Liu Q, Wang G. An anonymous offline RFID grouping-proof protocol. Future Internet 2018; 10(2): 1-15. doi:10.3390/fi10010002

[30] Tsai KY, Luo JN, Yang MH, Liew WT. Novel designated ownership transfer with grouping proof. Applied Sciences 2019; 9(4): 1-19. doi:10.3390/app9040724

[31] Luo JN, Yang MH. A Secure Partial RFID Ownership Transfer Protocol with Multi-Owners. Sensors 2020; 20(1): 1-18. doi: 10.3390/s20010022

[32] GS1. EPCTM Radio-Frequency Identity Protocols Generation-2 UHF RFID Specification for RFID Air Interface. Lawrenceville, NJ, USA: GS1 EPCglobal Inc, 2013.

[33] EPCGlobal GS1. EPC™ radio-frequency identity protocols generation-2 UHF RFID specification for RFID air interface protocol for communications at 860 MHz – 960 MHz version 2.0.1 ratified; 2015, pp. 1-152.

[34] Aydın Ö. Enhancing Security in RFID. PhD, Dokuz Eylül University, İzmir, Turkey, 2019.

[35] Dolev D, Yao A. On the security of public key protocols. IEEE Transactions on Information Theory 1983; 29(2): 198-208.

[36] Duan S, Mjølsnes SF, Tsay JK. Security analysis of the terrestrial trunked radio (TETRA) authentication protocol. In: Norwegian Information Security Conference; Stavanger; 2013. pp. 1-12.

[37] Rathore R, Hussain M. Simple, secure, efficient, lightweight and token based protocol for mutual authentication in wireless sensor networks. In: Shetty NR, Prasad NH, Nalini N (editors) Emerging Research in Computing, Information, Communication and Applications ERCICA 2015. New Delhi, India: Springer, 2015, pp. 451-462. doi: 10.1007/978-81-322-2550-8_43