International journal of innovation in Engineering

journal homepage: www.ijie.ir

Research Paper

SURC: Secure Ultra-lightweight RFID Authentication Protocol with Crossover

Mohammad Reza Mehrabani [1a] , **Soosan Sadegha** [b]

[a] *Dept. of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran*

[b] *Tehran University of Medical Science, Tehran, Iran.*

ARTICLE INFO

Received: 7 November 2022
Reviewed: 2 December 2022
Revised: 29 January 2023
Accepted: 16 February 2023

Keywords:

Radio frequency identification (RFID), security, privacy, authentication, confidentiality.

ABSTRACT

Radio Frequency Identification (RFID) system is a wireless automatic identification using low-cost RFID tags. Due to the importance of RFID in everyday life, the need to maintain security and Privacy in these systems has been increasing day by day. In this paper, at first, we define a new operation named crossover, by using simple bitwise operations and discuss its security. Then we propose a new secure ultra-lightweight RFID Authentication protocol with crossover operation (SURC). SURC is a low communication and computation cost protocol that can be integrated into the ubiquitous Electronic Product Code (EPCglobal) class1 Generation 2 tag protocol (C1G2). The new protocol resists data integrity and confidentiality, tag location tracking backward traceability, and server and tag impersonation.

1. Introduction

Identification technology through radio frequency (RFID) is a new technology that is used to identify objects and living creatures. Due to many advantages such as lower costs, higher speed and large-scale identification, the use of this technology has been increasing regularly. The basis of this technology is similar to barcode technology but in RFID systems, identification takes place without any physical contact or direct visibility and recognition perform via radio waves. These advantages make this technology attractive for commercial

[1] Corresponding author
Email Address: M.r.mehrabany@gmail.com

and industrial users, so that today the deployment of RFID systems in various areas such as credit cards, subways and buses, new Passport cards, and new electrical barcodes can be seen. Due to the use of RFID technology in important areas such as access control systems, supply chain management, new passports, non-contact smart cards, and etc., study on the security aspects of this technology and the deployment of security protocols for authentication in this system, is an urgent need for industries and organizations (Cole & Ranasinghe, 2008) (Mehrabani & Sadegha, 2021b) (Mehrabani & Sadegha, 2021a).

Most RFID frameworks use EPCglobal information and communication standards. EPC Class-1 Gen.-2 is a standard that is provided by EPCglobal organization (Duc, 2006). This standard gives a structure to RFID communications. EPC C-1 G-2 has confined labels to some straightforward math activities, for example, CRC (Cyclic Redundancy Checksum Code), PRNG (Pseudorandom Number Generator) and bitwise XOR (Chi-Fang, 2011). In this way, RFID authentication protocols dependent on EPC C-1 G-2 standard have gone through certain hardships to give amazing security viewpoints.

Authentication protocols are the most important tools to provide security in the highest layer, namely application layer, in RFID system. They run between the tag and card reader, and during their implementation, the parties review the accuracy of each other's identity to accept. Low cost production in RFID tags is one of the most important features (Shi et al., 2017), so there is no possibility of using encryption separately and security of the entire system depends directly on the security level of protocols that are used for identification and authentication.

In RFID systems, the most important features of a secure identification and authentication protocol are

- Confidentiality: Ensure that information is only available to those who are authorized to access this information (Shi et al., 2017). Confidentiality means that in a secure protocol there shouldn't be any leak-age of information to an attacker or eavesdrop-per. Information that is stored on the tag such as a unique identifier and secret keys that are employed in reconnaissance operations, should be only disposal to tag, and if necessary, the server should be final. Besides this, another point about confidentiality, is privacy and the location privacy of the tag or tag's holder. Location privacy of a tag means that, if an attacker, eavesdrop the session and save exchanged information during the identification protocol between the tag Ti and an allowed reader at time t, the attacker fails to identify any other transactions related to tag Ti, in the other time.
- Data integrity: Maintaining the authenticity and integrity of information and processing methods (Su et al., 2007). It means that no person should be able to alter or manipulate the exchange information between parties of protocol. Originality of the message, can prevent many attacks that lead to impersonation.
- Availability: Assurance that all authorized users can have access to information and other requirements (Bertolini et al., 2012). This means that authorized users (tags) at any time and any place need system services, be able to easily use them.
- Authentication: The most basic goal that an identifying protocol seeks, is checking the identity of an entity or parties of the protocol.

The remainder of this paper is divided into 5 parts. section 2 briefly reviews some recent RFID authentication protocols. Section 3 presents our new crossover operation and discuss its security features. Section 4, proposes a new authentication protocol with crossover operation (SURC), and while Section 5 discusses the security and the performance of the proposed protocol and compares it to the prior art, respectively. A few ends are introduced in section 6.

2. Related work

In 2009, Chen and Deng proposed a new mutual low-overhead authentication protocol that had 4 rounds (Chen & Deng, 2009). Their protocol relies on the use of PRNG primitives in tag and some other simple bitwise operations. Yeh et al. (2010) presented a mutual authentication protocol conforming to EPC Global class1 Gen-2 RFID Tag. Yeh et al.'s protocol uses pseudorandom number generator and simple bitwise operation and has six authentication steps (Yeh et al., 2010). Habibi et al, (2011) proved that does not assure the un-traceability and backward un-traceability aspects. Namely, all past and next transactions of a compromised tag will be traceable by an adversary (Habibi et al., 2011). Yoon (2012) pointed out that Yeh et al.'s protocol has serious security problems such as DATA integrity problem and forward secrecy problem (Yoon, 2012).

A mutual authentication protocol beneath the EPC C-1 G-2 standard was suggested by Chien & Chen (2007). They had utilized basic XOR, CRC, and PRNG in their plan (Chien & Chen, 2007). YI et al, (2012) showed some security problems of the protocol and proposed a new improved protocol (Yi et al., 2012). Peris -Lopez et al, (2009) showed some weaknesses of Chien and Chen's protocol including tag and reader impersonation and de-synchronization attack. They also showed that this protocol does not guarantee forward security and is vulnerable to trace attacks (Peris-Lopez et al., 2009). Han & Kwon (2009) also presented a de-synchronization attack and two tag impersonation attacks on Chien and Chen's protocol in new methods (Han & Kwon, 2009). These attacks were predominantly founded on frail weak secure properties of CRC.

Chien (2007) proposed a new ultra-lightweight RFID authentication protocol named SASI (Chien, 2007). The proposed scheme is ultra-lightweight, it has three shares of secret keys k1 and two random numbers nl, n2 by taking XOR operation to implement the encryption, in order to achieve forward security, the shared secret key, and the random number update each time. But because the key's updating does not adopt strict limits so this easily suffers a de-synchronization attack (Cao et al., 2009) (Castro et al., 2008) (Phan, 2009).

A security protocol with Only XOR and matrix operations was suggested by Karthikeyan & Nesterenko (2005) (Karthikeyan & Nesterenko, 2005). Phan (2009) showed that this protocol is at risk to some attacks like replay attacks and doesn't satisfy the un-traceability property (Phan, 2009). ARAP is a mutual authentication protocol that was proposed by Shen et al, (2010) (Shen et al., 2010). Niu et al, (2011), applied tag impersonation attack and de-synchronization attack on ARAP protocol (Niu et al., 2011). Wei et al. (2011) offered a mutual authentication protocol based on the hash function. In this protocol, the reader has its own identifier IDr and the backend server maintains old and new keys and also old and new random numbers (Wei et al., 2011). Niu et al, (2011), showed that Wei et al.'s (2011) protocol is vulnerable to Man-in-the middle attack (Niu et al., 2011).

3. Definition of the new operation

The tags in SURC use only three operations: bitwise XOR, Crossover operation Cros (A, B, C) and Pseudorandom Numbers Generators (PRNG). The crossover operation is defined as

3.1. Definition

Suppose A, B and C are L-bit strings, where

$$A = a_1 a_2 a_3 \dots a_l \; ; \quad a_i \in \{0,1\} , \qquad i = 1,2,3, \dots l \tag{1}$$

$$B = b_1 b_2 b_3 \dots b_l \; ; \quad b_j \in \{0,1\} , \qquad j = 1,2,3, \dots l$$

$$C = c_1 c_2 c_3 \dots c_l \; ; \quad c_k \in \{0,1\} , \qquad k = 1,2,3, \dots l$$

Then the crossover of A and C according to B denoted as $Cros(A,B,C)$ and is as

$$Cros(A,B,C) = (A \oplus PRNG(B)) \wedge B \vee (C \oplus PRNG(B)) \wedge Not(B) \qquad (2)$$

Where $\wedge$ is bitwise AND, $\vee$ is bitwise OR operations and $\oplus$ is bitwise XOR operation.

Suppose $Cros(A,B,C) = Cr_1 Cr_2 Cr_3 \dots Cr_l$ and $P = PRNG(B)$ where $P = p_1 p_2 p_3 \dots p_l$ where $p_i \in \{0,1\}$ and $i = 1,2,3, \dots l$. In order to compute $Cros(A,B,C)$, $P = PRNG(B)$ is computed, then based on string P, the participation of each a_i and c_i for producing output, is determined. For more details, if ith bit of the string B, b_i is 1, then Cr_i, i-th bit of the $Cros(A,B,C)$ will be $a_i \oplus p_i$ and if b_i is 0 then Cr_i will be replaced by $c_i \oplus p_i$. Fig. 1, shows the computation.

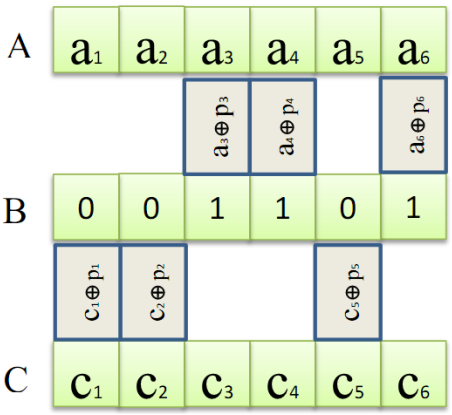

Fig. 1. Computation scheme of Crossover operation

Crossover operation can easily be implemented on passive tags. For implementation of crossover we need only to AND, OR, XOR and $PRNG(.)$ that can be used in EPC Class-1 Gen.-2 standard. The logical block diagram of the crossover operation is shown in Fig. 2.

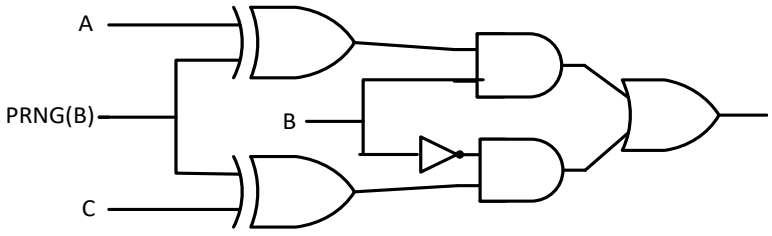

Fig. 2. The logical block diagram of the crossover operation

3.2. Security analyses of crossover operation

There are some remarks that should be noticed for analyses of this operation. First, there is no resemblance between the Hamming weight of the output string and inputs, so crossover can be used alone as it will not reveal any information about inputs Hamming weight. The other point is the effect of each sting on crossover's security that in the following we examine the effect of each string separately.

Now, suppose that the attacker has gained the string A and also has the output $Cros(A,B,C)$. We want to investigate that if knowing an A can reveal any information about B or C and threaten their privacy. The adversary cannot determine that which of a_i or c_i is used for producing Cr_i. In fact, to gain any information about B and C from A and output, an adversary should understand the position of A and C's entries in the output, and it is impossible.

1) Suppose that the attacker has gained the string C and also has the output $Cros(A,B,C)$. This situation is just like the above and having C will never threaten the privacy of A and B.
2) -Suppose that the attacker has gained the string A and also has the output $Cros(A,B,C)$. In this case, an adversary can easily compute $P = PRNG(B)$. If $m = W(p)$, then the adversary can find m entries of A and l-m entries of C, and this will threaten the privacy of A and C. To solve this problem, we can put some hidden strings such as EPC in this position and also XOR it with other hidden strings, so disclosing of EPC wouldn't be a threat to the other strings.
3) Suppose that the attacker has gained A and C and also has the output $Cros(A,B,C)$. This assumption can be investigated in two cases:

   Case 1: The i-th bit of A and C are different, $a_i \neq c_i$.

   In this case, an adversary will never find out the amount of p_i, because the adversary does not know which of a_i or c_i have been used to produce Cr_i.

   Case 2: The i-th bit of A and C are equal, $a_i = c_i$.

   In this case an adversary can easily find p_i. For example if $a_i = c_i = 0$ and $Cr_i = 1$, then p_i will be 1, and similarly in other three case p_i can be determined.

4) Suppose that A and C have m similar bits, so an adversary has m bit of P, where $P = PRNG(B)$. The question is that can knowing m bit of P reveals any information about the string B It is obvious that finding B from string P with m known bit, is as difficult as finding B from string P without any information about P, so knowing A and C will not reveal any information about B.

4. PROPOSED PROTOCOL

In this section we will present the proposed security protocol in detail.

4.1. Overview of Protocol

In the proposed protocol each tag stores a static identifier (EPC), an index-pseudonym and three keys K and P all of which are of 96 bit lengths to ensure compatibility with EPC Global encoding schemes. This information is also stored in a central database. To provide adequate protection from de-synch attacks, the backend database will store two tuples for each tag identifier: the current keys K and P and the last approved keys K and P. We first review the notations used in proposed protocol. Notations used in this paper are defined as

- *EPC*: The unique 96 bits identifier code in EPC Global encoding scheme.
- K_i: The authentication key stored in the tag for the database to authenticate the tag at the (© + 1)th

authentication phase.

- P_i: The access key stored in the tag for the tag to authenticate the database at the (© + 1)th authentication phase.
- K_{old} : The old authentication key stored in the database.
- K_{new} : The new authentication key stored in the database.
- P_{old} : The old access key stored in the database.
- P_{new} : The new access key stored in the database.
- X : The value kept as either new or old to show which key in the record of the database is found matched with the one of the tag.
- $A \rightarrow B$: A forwards a message to B.
- R_Y : The random number generated by device Y.
- $A \oplus B$: Message A is XOR with message B.

The information kept within respective devices:

- Tag: (K_i , P_i , EPC_s)
- Data-Base: (K_{old} , P_{old} , K_{new} , P_{new} , EPC)

Fig. 3 shows the proposed protocol which consists of two phases: the initialization phase, and the $(i + 1)$-th authentication phase.

4.2. Initialization phase

The manufacturer generates random values for K_0 and P_0 respectively, and sets the values for the record in the tag ($K_i = K_0$, $P_i = P_0$) and the corresponding record in the database ($K_{old} = K_{new} = K_0$, $P_{old} = P_{new} = P_0$).

4.2 The (i+1)-th authentication phase

Fig. 3 illustrates the $(i + 1)$-th authentication phase of proposed protocol. The detailed steps of the authentication phase are presented as follows:

- **Step 1. Reader $\rightarrow$ Tag:**

The reader generates random number as a challenge and forwards it to the tag.

- **Step 2. Tag $\rightarrow$ Reader: (A, B)**

After receiving R_r , the tag generates random number R_T , performs operations $A = cros(R_r, EPC \oplus R_T, K_i)$, $B = R_T \oplus K_i$, then forwards (A, B) back to the reader.

- **Step 3. Reader $\rightarrow$ Database: (A, B, Rr)**

The reader computes, forwards (A, B), received from the tag and R_r , generated in Step 1, to the database.

- **Step 4. Database $\rightarrow$ Reader: ©**

After receiving (A, B, R_r), the database performs the following operations:

1. Retrieves each stored RID sequentially $(K_{old}, P_{old}, K_{new}, P_{new}, EPS)$ to compute A with R_r , and compare the product with the received A to identify the correct matching record and authenticate the reader. To reach this goal database picks up an entry $(K_{old}, P_{old}, K_{new}, P_{new}, EPS)$ stored in itself, computes the values $l_{old} = cros(R_r, EPC \oplus B \oplus K_{old}, K_{old})$ and $l_{new} = cros(R_r, EPC \oplus B \oplus K_{new}, K_{new})$, and checks whether $l_{old} = A$ or $l_{new} = A$. The process is iteratively repeated for each entry until it finds a match. Once the matching record is found, set value X as old or new according to which authentication key in the record is found matched.
2. Server Computes $C = cros(P_x, R_T \oplus P_x, R_r)$, and forward them to the reader.
3. If X=new, then update the record by replacing K_{old} with K_{new} and P_{old} with P_{new}. New values for K_{new} and P_{new} will be reset as $PRNG(K_{new \oplus} P_{new})$ and $PRNG(P_{new} \oplus R_T)$.

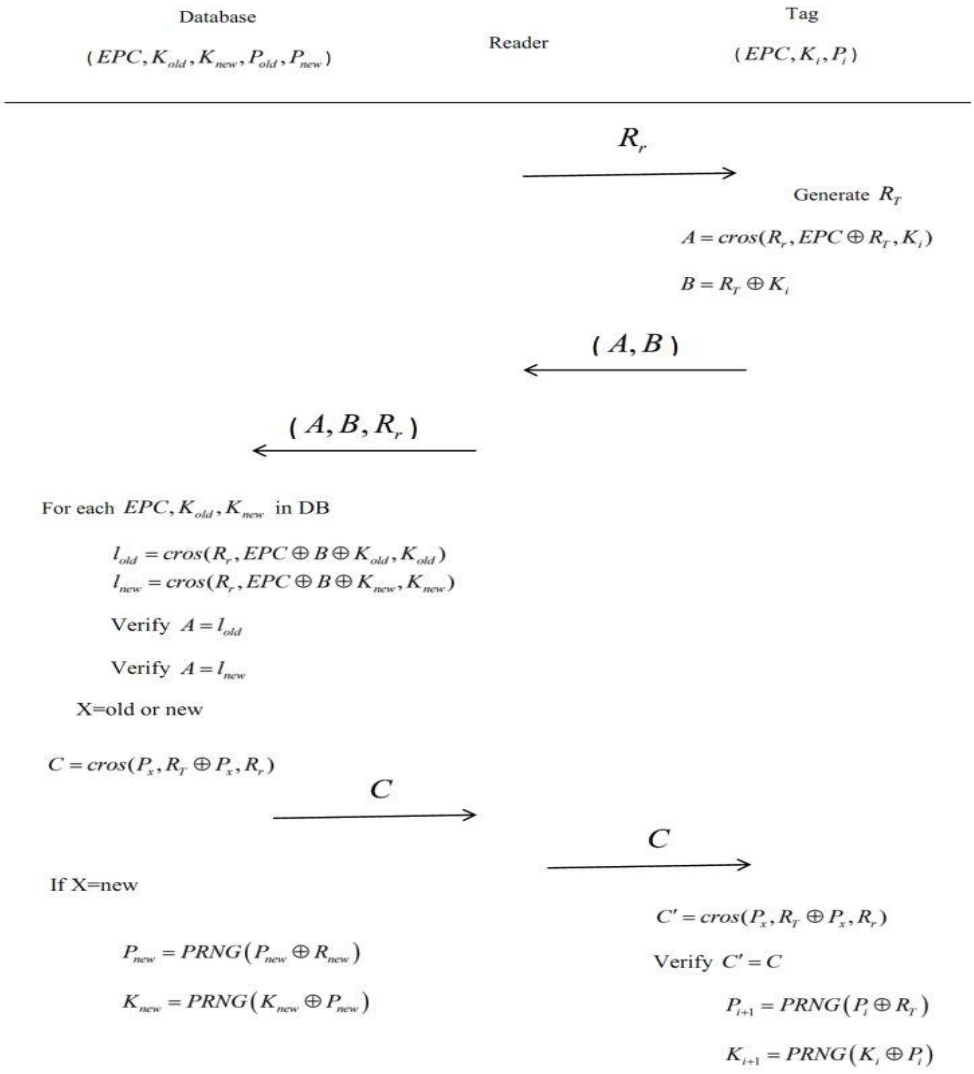

Fig. 3. The proposed $(i+1)$-th authentication phase

- **Step 5. Reader $\rightarrow$ Tag: C**

The reader forwards C to the tag.

- **Step 6.**

The tag computes $C' = cros(P_x, R_T \oplus P_x, R_r)$ and compares it with received C from reader. If they are equal then the authentication to the database is completed and the content kept inside is renewed as $K_{i+1} = PRNG(K_i \oplus P_i)$ and $P_{i+1} = PRNG(P_i \oplus R_T)$, for next access.

5. ANALYSIS

5.1. Security Analysis

The protocol has the following privacy and security properties.

1. **Mutual authentication**: The valid tag and the valid reader can authenticate each other. The massages A, B and C are all based on the shared keys K and P. Thus the valid party can generate these massages and be authenticated by the other party.
2. **Tag Information Privacy**: The detailed information K, P and EPC of tag is stored in database of the server, which is assumed to be secure. A server and a reader communicate via a secure channel. Only a legitimate server can extract a tag identifier from the massages. Also the Hamming weight of Crossover operation is not related to any of input, so the massages never leak any information about the tag and its secret numbers.

3. **Tag Location Privacy:** The responses of the tag T_i are anonymous. In fact, the eavesdropper cannot link tag responses to previous responses from the same tag, or distinguish one tag's responses from another's.
4. **Resistance to De-Synchronization Attack:** There are three different cases for creating de-synchronization between the tag and server, which are listed at below.
   - The server updates the tag entries while the tag does not update its secrets: Because the server holds old and new values of keys, if for any reason the server can updates keys, but the tag does not update its secrets, for example The attacker blocks the forwarding message which is sending from the reader to the tag at the step 5, de-synchronization will not happen, because in the next session tag will use the keys that are stored in the server as old keys.

     The tag updates its secrets but the server does not update the tag entries: In this protocol, because server's updating is done the keys in before than the tag, this case will never happen.
   - Both the tag and the server have updating, but with different values: This situation will happen only when the value of the generated R_T in tag be different from what the server has, but an attacker would never make such a difference because the R_T in the message A, has been used only to determine the position and an attacker cannot manipulate it.
5. **Tag Impersonation Attack:** An attacker cannot impersonate the tag, because for this goal he or she should have tag's secret keys. The attacker will never impersonate the tag without EPC and K, also he or she cannot attack when he or she have EPC or K alone and for this goal an attacker need to couple of them.
6. **Reply attack:** The protocol uses random numbers to resist reply attacks. The messages A, B and C are functions of freshly generated nonces R_T and R_r and so the messages cannot be used in other sessions.
7. **Backward security:** An attacker cannot identify the past interactions, even if it knows tag's present internal state. In fact, the attacker cannot detect tag's past interactions from its present state. It is obvious from $K_{i+1} = PRNG\left(K_i \oplus P_i\right)$ and $P_{i+1} = PRNG\left(P_i \oplus R_T\right)$ that attacker cannot easily find K_i and P_i from K_{i+1} and P_{i+1}.
8. **Server Impersonation Attack:** A legitimate server responds with a message C to tag in order to enable the tag to authenticate the server. Without knowing P, K and R_T an attacker cannot create valid C. An attacker would block round 5 and save sent C from server to the tag, but he will not be able to use this message in the other session, because in the next session the value of R_T will change and attacker cannot find it from A or B. so our protocol can resist impersonation attack.

In Table I, we compare our protocol with other lightweight protocols that have been recently proposed. It is clear from Table I that the proposed protocol, satisfy the greatest number of privacy and security properties.

Table 1. Security Analysis of SURC

property	Chien, (2007)	Chien & Chen, (2007)	Yeh et al, (2010)	Chen & Deng, (2009)	SURC
Information privacy	strong	vulnerable	vulnerable	strong	strong
Data confidentiality	strong	strong	vulnerable	strong	strong
Tag Impersonation	strong	vulnerable	strong	vulnerable	strong
Server Impersonation	strong	vulnerable	strong	vulnerable	strong
Backward Traceability	vulnerable	vulnerable	strong	vulnerable	strong
De-synchronization	vulnerable	vulnerable	strong	strong	strong

5.2. Efficiency analysis

In this chapter we analyze the efficiency of SURC protocol. To examine efficiency of the protocols we first

compare their computational costs. Table 2 shows that compare to the existing protocols, such as SASI, the proposed protocol does not suffer too much increase in computational cost. Storage requirement on the tag is the other factor that should be considered in analysis. This factor, as well as computational cost, is caused by memory limitations of the cheap price tag. The tag in SURC stores 3 strings: its unique EPC code, and two shared keys K and P. All the strings are L bits and so each tag needs storage of 3L bits. As we see in the table, compare to the other protocols, SURC requires less storage. Communication cost is the other factor that shows the efficiency of protocols. As we see in Table II, in SURC the tag transmits only two messages, hence our protocol in this respect, is one of the lightest protocols.

Table II. Efficiency analysis of SURC

property	Chien, (2007)	Chien & Chen, (2007)	Yeh et al, (2010)	Chen & Deng, (2009)	SURC
Operations	11 XOR 2 OR, 1 AND, 3+	2 XOR, 3Conjocate 3PRNG	4XOR 6PRNG	5XOR 2CRC	5XOR 5PRNG 4AND, 2OR, 2NOT
Storage requirement	7L	3L	4L	3L	3L
Communication message	2L	4L	3L	3L	2L

6. Conclusion

In this work, we have defined a new operation named Crossover. By using Crossover operation, we have proposed a new ultra-lightweight protocol that although it can solve the privacy and security problems of protocols, it is one of the lightest authentication protocols. In SURC there are only four operations for tags: bitwise AND, OR, XOR and PRNG. It has been compared with existing protocols with respect to both its privacy and security properties and its storage and computational requirements. The comparisons have shown that SURC is both more secure than other schemes and has some advantage over them, such as greatest number of security features and required less storage and computation in a tag.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Conflicts of Interest

The author declares no conflict of interest related to this publication.

References

- Bertolini, M., Ferretti, G., Montanari, R., Rizzi, A., & Vignali, G. (2012). A quantitative evaluation of the impact of the RFID technology on shelf availability. *International Journal of RF Technologies, 3*, 159-180. https://doi.org/10.3233/RFT-2012-019
- Cao, T., Bertino, E., & Lei, H. (2009). Security Analysis of the SASI Protocol. *IEEE Transactions on Dependable and Secure Computing*, *6*(1), 73-77. https://doi.org/10.1109/TDSC.2008.32
- Castro, J. C. H., Tapiador, J. E., Peris-López, P., & Quisquater, J.-J. (2008). Cryptanalysis of the SASI Ultralightweight RFID Authentication Protocol with Modular Rotations. *ArXiv*, *abs/0811.4257*.

- Chen, C.-L., & Deng, Y.-Y. (2009). Conformation of EPC Class 1 Generation 2 standards RFID system with mutual authentication and privacy protection. *Engineering Applications of Artificial Intelligence*, *22*(8), 1284-1291. https://doi.org/https://doi.org/10.1016/j.engappai.2008.10.022
- Chi-Fang, H. (2011). Low-Cost Solution for RFID Tags in Terms of Design and Manufacture. In T. Cornel (Ed.), *Current Trends and Challenges in RFID* (pp. Ch. 6). IntechOpen. https://doi.org/10.5772/17257
- Chien, H.-Y., & Chen, C.-H. (2007). Mutual authentication protocol for RFID conforming to EPC Class 1 Generation 2 standards. *Computer Standards & Interfaces*, *29*(2), 254-259. https://doi.org/10.1016/j.csi.2006.04.004
- Chien, H. Y. (2007). SASI: A New Ultralightweight RFID Authentication Protocol Providing Strong Authentication and Strong Integrity. *IEEE Transactions on Dependable and Secure Computing*, *4*(4), 337-340. https://doi.org/10.1109/TDSC.2007.70226
- Cole, P. H., & Ranasinghe, D. C. (2008). *Networked RFID Systems and Lightweight Cryptography*. https://doi.org/10.1007/978-3-540-71641-9
- Duc, D. N., Park, J., Lee, H., & Kim, K. . (2006). *Enhancing security of EPCglobal Gen-2 RFID tag against traceability and cloning* Symposium on Cryptography and Information Security, Hiroshima, Japan.
- Habibi, M., Gardeshi, M., & Alaghband, M. R. (2011). Practical Attacks on a RFID Authentication Protocol Conforming to EPC C-1 G-2 Standard. *International Journal of UbiComp*, *2*(1), 1-13. https://doi.org/10.5121/iju.2011.2101
- Han, D., & Kwon, D. (2009). Vulnerability of an RFID authentication protocol conforming to EPC Class 1 Generation 2 Standards. *Computer Standards & Interfaces*, *31*(4), 648-652. https://doi.org/https://doi.org/10.1016/j.csi.2008.06.006
- Karthikeyan, S., & Nesterenko, M. (2005). *RFID security without extensive cryptography* Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks, Alexandria, VA, USA. https://doi.org/10.1145/1102219.1102229
- Mehrabani, M., & Sadegha, S. (2021a). Improvement of the Efficient Mutual Authentication Protocol for Passive RFID Tags (EMAP). *International Journal of Innovation in Engineering*, *1*(3), 1-7. https://doi.org/10.59615/ijie.1.3.1
- Mehrabani, M., & Sadegha, S. (2021b). Security Analysis and Improvement of Wei-Chi Ku and Yi-Han Chen's RFID protocol. *International Journal of Innovation in Engineering*, *1*(2), 73-83. https://doi.org/10.52547/ijie.1.2.73
- Niu, B., Li, H., Zhu, X., & Lv, C. (2011, 3-4 Dec. 2011). Security Analysis of Some Recent Authentication Protocols for RFID. 2011 Seventh International Conference on Computational Intelligence and Security, https://doi.org/10.1109/CIS.2011.152
- Peris-Lopez, P., Hernandez-Castro, J. C., Estevez-Tapiador, J. M., & Ribagorda, A. (2009). Cryptanalysis of a novel authentication protocol conforming to EPC-C1G2 standard. *Computer Standards & Interfaces*, *31*(2), 372-380. https://doi.org/10.1016/j.csi.2008.05.012
- Phan, R. C. W. (2009). Cryptanalysis of a New Ultralightweight RFID Authentication Protocol—SASI. *IEEE Transactions on Dependable and Secure Computing*, *6*(4), 316-320. https://doi.org/10.1109/TDSC.2008.33
- Shen, J., Choi, D., Moh, S., & Chung, I. (2010, 4-6 Nov. 2010). A Novel Anonymous RFID Authentication Protocol Providing Strong Privacy and Security. 2010 International Conference on Multimedia Information Networking and Security, https://doi.org/10.1109/MINES.2010.128
- Shi, Z., Chen, J., Chen, S., & Ren, S. (2017, 25-26 March 2017). A lightweight RFID authentication protocol with confidentiality and anonymity. 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), https://doi.org/10.1109/IAEAC.2017.8054290
- Su, W., Alchazidis, N., & Ha, T. T. (2007, 29-31 Oct. 2007). Data Integrity in RFID Systems. MILCOM 2007 - IEEE Military Communications Conference,
- Wei, C. H., Hwang, M. S., & Chin, A. Y. h. (2011). A Mutual Authentication Protocol for RFID. *IT Professional*, *13*(2), 20-24. https://doi.org/10.1109/MITP.2011.17
- Yeh, T.-C., Wang, Y.-J., Kuo, T.-C., & Wang, S.-S. (2010). Securing RFID systems conforming to EPC Class 1 Generation 2 standard. *Expert Systems with Applications*, *37*(12), 7678-7683. https://doi.org/https://doi.org/10.1016/j.eswa.2010.04.074
- Yi, X., Wang, L., Mao, D., & Zhan, Y. (2012). An Gen2 Based Security Authentication Protocol for RFID System. *Physics Procedia*, *24*, 1385-1391. https://doi.org/10.1016/j.phpro.2012.02.206
- Yoon, E.-J. (2012). Improvement of the securing RFID systems conforming to EPC Class 1 Generation 2 standard. *Expert Systems with Applications*, *39*(1), 1589-1594. https://doi.org/https://doi.org/10.1016/j.eswa.2011.07.053