On the Security of RFID Anti Cloning Security Protocol(ACSP)

Masoumeh Safkhani[1], Nasour Bagheri[2] and Majid Naderi[1]

[1] Electrical Engineering Department, Iran University of Science and Technology, Tehran, Iran.
{M_Safkhani,M_Naderi}@iust.ac.ir
[2] Electrical Engineering Department, Shahid Rajaee Teacher Training University, Tehran, Iran.
NBagheri@srttu.edu

Abstract. Recently Qian **et al.** [38] have proposed a new attack for RFID systems, called counting attack, where the attacker just aims to estimate the number of tagged objects instead of steal the tags' private information. They have stated that most of the existing RFID mutual authentication protocols are vulnerable to this attack. To defend against counting attack, they propose a novel Anti-Counting Security Protocol called ACSP. The designers of ACSP have claimed that their protocol is resistant against counting attack and also the other known RFID security threats. However in this paper we present the following efficient attacks against this protocol:

- Tag impersonation attack: the success probability of attack is "1" while the complexity is two runs of protocol.
- Two single tag de-synchronization attacks, the success probability of both attacks are "1" while the complexity is at most two runs of protocol.
- Group of tags de-synchronization attack: this attack, which can de-synchronize all tags in the range at once, has success probability of "1" while its complexity is one run of protocol. .
- Traceability attack: the adversary's advantage in this attack is almost $\frac{1}{2}$, which is almost the maximum of possible advantages for an adversary in the same model. The complexity of attack is three runs of protocol

Keywords: RFID, Authentication, Counting attack, ACSP, Tag Impersonation Attack, De-synchronization Attack, Traceability Attack.

1 Introduction

Radio Frequency Identification (RFID) systems consist of a reading device called reader and one or more tags. The reader is a powerful device which can read/modify the tag's information. Tags are very constraint devices that range from passive tags, which respond only at reader commands, to active tags, which have an on-board power supply.

The design of secure authentication protocols for low-cost RFID tags has received the attention of a lot of researchers, though many protocols have been published lately [1, 4, 6–12, 15–19, 21, 22, 27, 28, 30–37]. However, most of them have not satisfied the claimed security goles [1–3, 5, 13, 14, 19, 20, 23–30]. The security of an RFID protocol can be analyzed in several direction. For example, an adversary may be interested in tracking the tag holder which can compromise the tag holder privacy or it may try to clone a legitimate tag to pay less. We can classify the main known attacks in the field of RFID as follows:

- Forgery attacks that include:
  - Tag impersonation
  - Reader impersonation
- Secret disclosure attacks that include:
  - Reader's secret values, e.g. shred key, disclosure attack
  - Tag's secret values, e.g. tag's ID, disclosure attack
- De-synchronization attacks
- Traceability attacks

– Cloning attacks
– Replay attacks

Recently, Qian **et al.** have proposed a new attack, called counting attack, in which the adversary's target is to estimate the number of tagged objects. They have stated that most of the existing RFID protocols are vulnerable to counting attack. In addition, to defend against this attack, they have proposed a new protocol entitled ACSP [38]. Qian **et al.** have claimed that ACSP is secure against counting attack and also against the other attacks in the context. However, we show that their claims do not hold.

Paper Contributions: In this paper we show three kinds of attacks against ACSP. The first attack is the tag impersonation attack which is able to forces the reader to authenticate the adversary as a legitimate tag. The second attack is a de-synchronization attack which is able to de-synchronize the communication between the tag and the reader and the third attack is a traceability attack which is able to trace tags and compromise the privacy of the tag holder. The success probability of all attacks is "1", the complexity of any given attack is at most two runs of protocol and the main computation cost of any attack is at most one calculation of hash function and one calculation of CRC function.

Paper Organization: The notations used in the paper are presented in Section 2. The ACSP protocol is briefly described in Section 3. In Section 4 we describe our impersonation attack which can be considered as de-synchronization attack also. We present another de-synchronization attack which is de-synchronize a single tag and the reader in Section 5. In Section 6, we present another de-synchronization which de-synchronizes all tages in the range. Traceability attack is described in Section 7. Finally, we conclude the paper in Section 8.

2 Preliminaries

Throughout the paper, we use the following notations:

- R: RFID reader.
- T: RFID tag.
- SID: The session identifier.
- SID_{cur}: The current session identifier.
- SID_{new}: The new session identifier used in the following communication session.
- TID : The current unique identifier of tag.
- TID_{new}: The next identifier of tag used in the following communication session.
- $R_i, 1 \leq i \leq 5$: Pseudo random numbers that are generated by the reader or the tag.
- $H(x_1, x_2)$: One way hash function with variables x_1 and x_2.
- CRC: Cyclic Redundancy Code.
- $MASKVAL$: Bit mask.
- $\oplus$: Exclusive-or operation.
- $\mathcal{A}$: Adversary.
- $\overline{MSG_HEADER}$: Header of message which indicates the type of the message, including Select, Query, Identification, Authentication and End commands.

- $\overline{SELECT}$: Select command.
- $\overline{QUERY}$: Query command.
- $\overline{IDENT}$: identification message from tag, containing TID information.
- $\overline{AUTHEN}$: Authentication message.
- $\overline{UERYREP}$: End current slot command.
- $\overline{QUERYADJUST}$: End current slot command.

Each message which is sent from the reader to the tag or wise versa includes a $\overline{MSG_HEADER}$, the message body and the CRC of the message. The $\overline{MSG_HEADER}$ can be $\overline{SELECT}$, $\overline{QUERY}$, $\overline{IDENT}$, $\overline{AUTHEN}$, $\overline{UERYREP}$ or $\overline{QUERYADJUST}$.

3 ACSP Description

Recently Qian *et al.* [38] have proposed a new attack for RFID system which has been called counting attack. The attacker's goal in counting attack is to estimate the number of tagged objects instead of stealing private information of tags. They have stated [38] that most of the existing RFID mutual authentication protocols are vulnerable to counting attack. To defend against this attack, they have proposed a novel anti-counting security protocol or in short term called ACSP [38]. In this section we present a brief description of ACSP.

ACSP is composed of two phases: SID Update and Tag Identification that described separately in below:

SID Update Phase: This phase of protocol, which is depicted in Fig.2, accomplishes as below:

1. The reader generates two random numbers R_1 and R_2 and sends $(\overline{UPDSID}, R_1, R_2, H(R_1, SID_{cur}), CRC)$ to the tag.
2. Upon receipt of the message, each tag checks the values of $H(R_1, SID_{cur})$ and CRC to verify whether the command is valid. If it is valid, tags update local SID as $SID_{new} = H(R_1 \oplus R_2, SID_{cur})$.

Tag Identification Phase: This phase of protocol, which is depicted in Fig. 1, works as follows:

1. The reader R generates a random number R_3 and sends $(\overline{SELECT}, R_3,\ H(R_3, SID), (MASKVAL \oplus SID), CRC)$, to the target tag.
2. Upon receipt of the message, any tag T_i checks the included CRC, R_3 and corresponding $H(R_3, SID)$. If all are correct, the tag extracts $MASKVAL$ by calculating $(MASKVAL \oplus SID) \oplus SID$. If the tag's ID matches $MASKVAL$, the tag gets ready to respond, otherwise the tag keeps silent until receiving the next Select command.
3. R generates a random number R_4 and sends $(\overline{QUERY},\ R_4, H(R_4, SID), CRC)$.
4. Upon receipt of the message, a ready tag T_i checks CRC and correctness of $(R_4, H(R_4, SID))$ in the Query command. If they are correct, it generates a random number R_5 and responds with $(\overline{IDENT}, R_5, H(R_4, TID), CRC)$ as identification message.
5. Upon receipt of the identification message without collision, R proceeds as follows:
   (a) It will search for the tag's TID in its database according to R_4 and $H(R_4, TID)$.
   (b) If R finds the TID of the tag it does as follows:
      - Updates tag's TID as $TID_{new} = H(R_4 \oplus R_5, TID)$.
      - Sends $(\overline{AUTHEN}, H(R_5, TID), CRC)$ to the tag.
   (c) Else, the reader sends $(\overline{QUERYREP}/\overline{QUERYADJUST}, R_p, H(R_p, SID), CRC)$ to end this slot and start a new one.
6. The tag receives $(\overline{AUTHEN}, H(R_5, TID), CRC)$ and upon receipt of these values, it checks $H(R_5, TID)$ and proceeds as follows:
   (a) If it is correct, it update its ID as $ID = H(R_4 \oplus R_5, TID)$.
   (b) Else, , it does nothing.

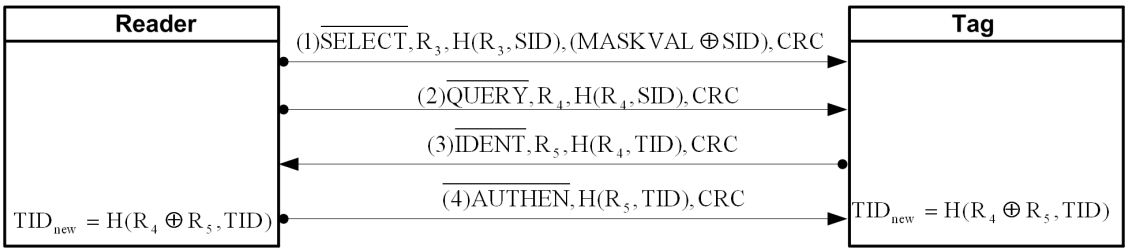

Fig. 1. The ACSP's Tag Identification Phase.

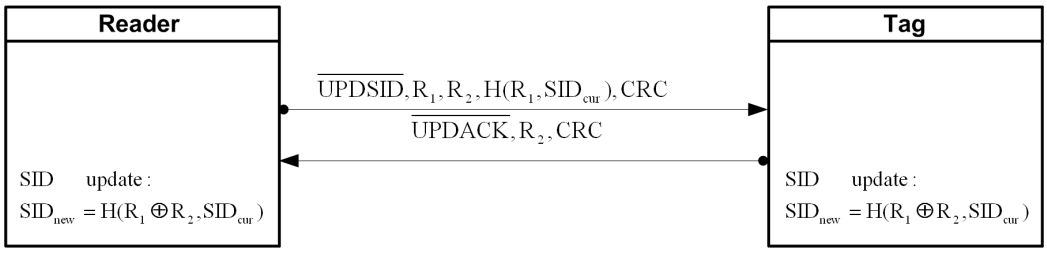

Fig. 2. The ACSP's SID Update Phase.

4 Tag Impersonation Attack

Tag impersonation attack is a forgery attack in which the reader accepts a spoofed tag as a legitimate tag. Any secure RFID authentication protocols must resistance against all kind of forgery attacks, include tag impersonation attack. In this section, we prove that ACSP is vulnerable to tag impersonation attack. In our tag impersonation attack, to impersonate the tag T_i the adversary ($\mathcal{A}$) can follow the bellows steps:

Phase 1: Retrieving TID_{new}

1. The reader R generates a random number R_3 and sends $(\overline{SELECT}, R_3,\ H(R_3, SID),\ (MASKVAL \oplus SID), CRC)$, to the target tag T_i. $\mathcal{A}$ does not change the transferred message of this step.
2. Upon receipt of the message, any tag T_j checks the included CRC, R_3 and corresponding $H(R_3, SID)$. If all are correct, the T_i extracts $MASKVAL$ by calculating $(MASKVAL \oplus SID) \oplus SID$. If the tag's ID matches $MASKVAL$, T_i gets ready to respond, otherwise it keeps silent until receiving the next Select command. $\mathcal{A}$ does not change the transferred message of this step.
3. R generates a random number R_4 and sends $(\overline{QUERY},\ R_4, H(R_4, SID), CRC)$. $\mathcal{A}$ does not change the transfered message of this step.
4. Upon receipt of the message, a ready tag T_i checks CRC and correctness of $(R_4, H(R_4, SID))$ in the Query command. If they are correct, it generates a random number R_5 and responds with $(\overline{IDENT}, R_5, H(R_4, TID), CRC)$ as identification message.
5. $\mathcal{A}$ intercepts $(\overline{IDENT}, R_5, H(R_4, TID), CRC)$, changes the R_5 value to zero, computes the corresponding CRC of $\overline{IDENT}, 0, H(R_4, TID)$ and sends$(\overline{IDENT}, 0, H(R_4, TID), CRC)$ to the reader.
6. Upon receipt of the identification message without collision, R proceeds as follows:
   (a) It will search for the tag's TID in its database according to R_4 and $H(R_4, TID)$.
   (b) If R finds the TID of T_i, which will find, it does as follows:
   - Updates T_i's TID as $TID_{new} = H(R_4 \oplus R_5, TID) = H(R_4 \oplus 0, TID) = H(R_4, TID)$.
   - Sends $(\overline{AUTHEN}, H(R_5, TID), CRC)$ to T_i.

7. $\mathcal{A}$ blocks this message.

Hence, following the above procedure, the adversary knows the $TID_{new} = H(R_4, TID)$, the secret TID of T_i stored in the reader database.

Phase 2: Tag impersonation : to impersonate the tag, $\mathcal{A}$ waits until the reader initiates a new session to identify any tag $T_j$where:

1. R generates a random number R_3 and sends $(\overline{SELECT}, R_3, H(R_3, SID), (MASKVAL \oplus SID), CRC)$.
2. Upon receipt of the message, $\mathcal{A}$ gets ready to respond.
3. R generates a random number R_4 and sends $(\overline{QUERY}, R_4, H(R_4, SID), CRC)$.
4. Upon receipt of the message, $\mathcal{A}$ blocks any responds from other tags towards R and generates a random number R_5 and responds with $(\overline{IDENT}, R_5, H(R_4, TID), CRC)$ as identification message, where TID has has been retrieved from **Phase 1** of attack.
5. Upon receipt of the identification message without collision, R proceeds as follows:
   (a) It will search for the tag's TID in its database according to R_4 and $H(R_4, TID)$.
   (b) It finds the TID of T_i and does as follows:
   - Updates T_i's TID as $TID_{new} = H(R_4 \oplus R_5, TID) = H(R_4 \oplus 0, TID) = H(R_4, TID)$.
   - Sends $(\overline{AUTHEN}, H(R_5, TID), CRC)$ to T_i.
6. $\mathcal{A}$ updates its record of TID to $TID_{new} = H(R_4 \oplus R_5, TID)$.

So the reader authenticates adversary as a legitimate tag and updates tag's TID. The success probability of above attack is "1" and the complexity is only two successive runs of protocol. We emphasis that, as it has been indicated in the attack, on **Phase 2** of the attack the adversary does not need to wait for an special session of protocol between R and the target tag T_i to impersonate T_i but it can impersonate T_i on any run of protocol after **Phase 1** of attack, where the adversary retrieves TID of T_i.

Remark 1. After the successful run of the given tag impersonation attack, the reader authenticates the adversary as a legitimate tag and updates tag's TID while the legitimate tag has not updated its TID. Hence, the reader and the legitimate tag will not authenticate each other anymore in the following transactions. Therefore, the given impersonation attack leads to de-synchronization attack.

Remark 2. To overcome the potential de-synchronization attack because of the message lost problem occurring in the tag identification procedure, designers suggested [38, Sec. 6] that the reader preserves a copy of TID used in the last successful identification for each tag. However, this modification also does not improve the security of the protocol against the proposed attack.

5 Single Tag De-synchronization Attack

In de-synchronization attack, the adversary forces the tag and the reader to update their common values to different values from each other. If the adversary can be succeed in forcing the tag and the reader to do so, they will not authenticate each other in the further transactions. Qian *et al.* [38] have stated that it is possible to desynchronize the tag and the reader in ACSP if the adversary block the transferred message from the reader to the tag in step 5b and to solve the problem , as an enhanced protocol, they suggested [38, Sec. 6] that the reader preserves a copy of TID used in the last successful identification for each tag. However, in section 4 we have described a powerful tag impersonation attack which also desynchronize the target tag and the reader in their original and enhanced protocols.

In this section, we present a different de-synchronization attack against the original protocol which is not follow Qian *et al.*'s suggestion to attack the protocol. Our de-synchronization attack is rather simple and it is based on this fact that in step 3 of the ACSP protocol the random number R_5 does not effect any part of the transfered message, except the CRC value. Hence, if

an adversary $\mathcal{A}$ changes R_5 and CRC properly, there is now way for the tag to understand the modification. We use this observation in our de-synchronization attack against ACSP. In this attack, $\mathcal{A}$ does as follows:

- $\mathcal{A}$ lets step 1 and step 2 of protocol to be run without any change.
- $\mathcal{A}$ intercepts the message of step 3 of protocol and changes the value of R_5 to some arbitrary value, i.e. $R_5 \oplus \Delta$, and computes the corresponding CRC of $(\overline{IDENT}, R_5 \oplus \Delta, H(R_4, TID))$ denoted by CRC'. Then it sends $(\overline{IDENT}, R_5 \oplus \Delta, H(R_4, TID), CRC')$ to the reader.
- The reader authenticates the man in the middle adversary as a legitimate tag. Then R updates tag's TID as $TID_{new} = H(R_4 \oplus R_5 \oplus \Delta, TID)$.

It must be noted that the reader updates the tag's TID by $R_5 \oplus \Delta$ while the tag does not update its TID so they cannot authenticate each other in the following transactions. The success probability of above attack is "1" and the complexity is only one run of protocol.

6 Group De-synchronization Attack

In this section we introduce an attack which is desynchronized all tags in the range. It must be noted that in ACSP the reader and tags share the session identifier, SID. This value gets updated periodically following the SID Update Phase 3 of protocol. However, if an adversary forces the tag to updates its SID_{new} to a different value compared to the value in the reader side, then the Tag Identification Phase 3 of protocol can not be run properly and the tag and the reader will be desynchronized. In addition, in SID Update Phase of protocol, all tags must update their SID, otherwise they will lost their synchronization with the reader. On the other hand, to update SID, the reader generates two random numbers R_1 and R_2 and sends $(\overline{UPDSID}, R_1, R_2, H(R_1, SID_{cur}), CRC)$ to tags. Each tag, upon receipt of the message, checks the values of $H(R_1, SID_{cur})$ and CRC and if it is valid it updates local SID as $SID_{new} = H(R_1 \oplus R_2, SID_{cur})$. It can be seen that the adversary $\mathcal{A}$ can do as follows:

1. $\mathcal{A}$ intercepts the message,
2. changes R_2 to $R_2' \neq R_2$,
3. calculates the $CRC' = CRC(\overline{UPDSID}, R_1, R_2, H(R_1, SID_{cur}))$,
4. and broadcasts $(\overline{UPDSID}, R_1, R_2, H(R_1, SID_{cur}), CRC)$ to all tags in the range.

Obviously, the modified message will pass the tags verification test and all tags in the range will update their SID to $SID_{new} = H(R_1 \oplus R_2', SID_{cur})$. Since with a high probability $H(R_1 \oplus R_2, SID_{cur}) \neq H(R_1 \oplus R_2', SID_{cur})$ (the exact probability is $1 - 2^{-n}$ where n is length of SID in bits), all tags are desynchronized from the reader. The success probability of above attack is almost "1" and the complexity is only one run of protocol.

Remark 3. It must be noted that the designers have discussed [38, Sec. 6] that the the tag and the reader will be de-synchronized if tag does not receive the $\overline{UPDSID}$ command. To overcome this problem they suggested any tag to send a update acknowledge command as $(\overline{UPDACK}, R2, CRC)$ to the reader. However, this message can be generated by the adversary and sent to the reader because it does not include any secret parameter. Therefore, the enhanced protocol is also vulnerable to the given attack.

7 Traceability Attack

In this section we show how ACSP puts at stake the location privacy of tags' holders because tags can be tracked with a high probability. Specifically, adversary is given a target tag T_i which is supposed to trace. Later, a random tag T_j is given to $\mathcal{A}$ and the adversary should verify whether it is T_i. It output its decision as a single bit $\widetilde{b} \in \{0, 1\}$, 0 for to $T_j \neq T_1$ and

1 for to $T_j = T_1$ cases. A's success in winning the traceability game G is equivalent to the success of breaking the untraceability property offered by the protocol. So the advantage of A in distinguishing whether the messages correspond to T_i or not is defined as below:

$$Adv_{\mathcal{A}}^{\mathsf{UNT}}(q, kr) = |Pr[\widetilde{b}iscorrect] - \frac{1}{2}|$$

where q is a security parameter (i.e. the bit length of the key shared between the tag and the reader) and kr is the number of times $\mathcal{A}$ runs the protocol. Our traceability attack is described as below:

Phase 1: Learning , in this phase of attack the adversary de-synchronize T_i and R. In addition, though the de-synchronization attack it can achive the required information to trace T_i later.

1. $\mathcal{A}$ desynchronize T_i and R following the given tag impersonation attack which also leads to de-synchronization(refer to 1) and store the following messages transfered between R to T_i, include:
   - $(\overline{SELECT}, R_3,\ H(R_3, SID),\ (MASKVAL \oplus SID), CRC)$, from R to T_i.
   - $(\overline{QUERY},\ R_4, H(R_4, SID), CRC)$,from R to T_i.
   - $(\overline{IDENT}, R_5, H(R_4, TID), CRC)$,from T_i to R.

Phase 2: Challenge , in this phase of protocol the adversary does as follows:

1. $\mathcal{A}$ sends the eavesdropped $(\overline{SELECT}, R_3,\ H(R_3, SID),\ (MASKVAL \oplus SID), CRC)$ to the target tag T_j.
2. Upon receipt of the message, T_j checks the included CRC, R_3 and corresponding $H(R_3, SID)$. If $T_i = T_j$ then all are correct and T_i extracts $MASKVAL$ by calculating $(MASKVAL \oplus SID) \oplus SID$. The tag's ID matches $MASKVAL$, T_i gets ready to respond. If $T_i \neq T_j$ it will keep silence.
3. $\mathcal{A}$ sends the eavesdropped $(\overline{QUERY},\ R_4, H(R_4, SID), CRC)$ to T_j.
4. Upon receipt of the message, if T_j is ready, it checks CRC and correctness of $(R_4, H(R_4, SID))$ in the Query command which. If they are correct then it generates a random number R_5' and responds with $(\overline{IDENT}, R_5', H(R_4, TID'), CRC')$ as identification message.
5. $\mathcal{A}$ eavesdropped $(\overline{IDENT}, R_5', H(R_4, TID'), CRC')$ and go to the next phase of attack.

Phase 3: Guessing , $\mathcal{A}$ finishes $\mathcal{G}$ and outputs a bit $\widetilde{b}$ as its conjecture of the value b. In particular, $\mathcal{A}$ utilizes the following simple decision rule in generating the decision bit $\widetilde{b}$:

$$\begin{cases} \text{if } H(R_4, TID) == H(R_4, TID') & \widetilde{b} = 1 \\ \text{if} \quad X \neq Y & \widetilde{b} = 0 \end{cases} \tag{1}$$

In the given attack, the adversary desynchronize the reader and the target tag T_i at the first phase of attack. Hence, T_i will not update its secret parameters include SID and TID. On the other hand, as long as SID and TID are fixed, the eavesdropped values in the Learning phase of attack are pass the tags verification tests in the protocol and the adversary will expect the same $H(R_4, TID)$ from the tag to be included in its $\overline{IDENT}$ command. Hence, if $T_i = T_j$ then with the probability of "1" we have $H(R_4, TID) == H(R_4, TID')$. However, if $T_i \neq T_j$ then with a negligible probability $H(R_4, TID) == H(R_4, TID')$. This probability is less than 2^{-n}, where n is the output length of hash function, $H(.)$. Therefore, the success probability of adversary to output the correct $\widetilde{b}$ is lower bounded by $1 - 2^{-n}$. Hence, the adversary advantage in wining the game is as follows, which is almost the maximum of possible advantages:

$$Adv_{A}^{\mathsf{UNT}}(q, 1) = |1 - 2^{-n} - \frac{1}{2}| = |\frac{1}{2} - 2^{-n}|$$

The total complexity of attack is "3" runs of protocol, two runes in Learning phase and one run in Challenge phase of attack.

8 Conclusion

In this paper we considered the ACSP, UHF RFID mutual authentication protocol. Based on its designers' climes this protocol is supposed to resist counting attack and the other known attacks against an RFID system. However, we presented several efficient attack against this protocol with high success probability. Our attacks include tag impersonation attack, single tag de-synchronization attack, group of tags de-synchronization attack and traceability attack. The success probability of all attacks are almost "1" and the complexity are at most three runs of the protocol. Hence, ACSP is not a secure protocol for an ordinary applications of RFID systems.

References

1. N. Bagheri, P. Peris-Lopez, M. Safkhani, M. Naderi, and J. C. Hernandez-Castro. On the Security of Tan *et al.* Serverless RFID Authentication and Search Protocols. Manuscript, 2011.
2. N. Bagheri, M. Safkhani, M. Naderi, and S. K. Sanadhya. Security Analysis of $LMAP^{++}$, an RFID Authentication Protocol. Cryptology ePrint Archive, Report 2011/193, 2011. `http://eprint.iacr.org/`.
3. M. Bárász, B. Boros, P. Ligeti, K. Lója, and D. Nagy. Passive Attack Against the M2AP Mutual Authentication Protocol for RFID Tags. In *First International EURASIP Workshop on RFID Technology*, Vienna, Austria, September 2007.
4. M. Burmester, B. de Medeiros, J. Munilla, and A. Peinado. Secure EPC gen2 compliant radio frequency identification. In P. M. Ruiz and J. J. Garcia-Luna-Aceves, editors, *ADHOC-NOW*, volume 5793 of *Lecture Notes in Computer Science*, pages 227–240. Springer, 2009.
5. T. Cao, E. Bertino, and H. Lei. Security Analysis of the SASI Protocol. *IEEE Trans. Dependable Sec. Comput.*, 6(1):73–77, 2009.
6. H.-Y. Chien and C.-H. Chen. Mutual authentication protocol for RFID conforming to EPC class 1 generation 2 standards. *Computer Standards & Interfaces*, 29(2):254–259, 2007.
7. J.-S. Cho, S.-S. Yeo, and S. K. Kim. Securing against brute-force attack: A hash-based RFID mutual authentication protocol using a secret value. volume 34, pages 391–397, 2011.
8. E. Y. Choi, D. H. Lee, and J. I. Lim. Anti-cloning protocol suitable to EPCglobal class-1 generation-2 RFID systems. *Computer Standards & Interfaces*, 31(6):1124–1130, 2009.
9. Class-1 generation 2 UHF air interface protocol standard version 1.2.0, Gen2, 2008. `http://www.epcglobalinc.org/standards/`. In *Gen-2 Standard*. EPCGlobal, 2008.
10. EPC Tag data standar dversion 1.4.2008. http://www.epcglobalinc.org/standards/. Yearly report on algorithms and keysizes, Technical Report D.SPA.13Rev.1.0,ICT-2007-216676,. In *Gen2 Standard*. ECRYPT, 2010.
11. C. Hung-Yu. SASI: A New Ultralightweight RFID Authentication Protocol Providing Strong Authentication and Strong Integrity. *IEEE Transactions on Dependable and Secure Computing*, 4(4):337–340, December 2007.
12. T. Li. Employing Lightweight Primitives on Low-Cost RFID Tags for Authentication. In *VTC Fall*, pages 1–5. IEEE, 2008.
13. T. Li and R. H. Deng. Vulnerability Analysis of EMAP - An Efficient RFID Mutual Authentication Protocol. In *Second International Conference on Availability, Reliability and Security - AReS 2007*, Vienna, Austria, April 2007.
14. T. Li and R. H. Deng. Vulnerability analysis of EMAP-an efficient RFID mutual authentication protocol. In *ARES*, pages 238–245. IEEE Computer Society, 2007.
15. N.-W. Lo and K.-H. Yeh. An efficient mutual authentication scheme for EPCglobal class-1 generation-2 RFID system. In M. K. Denko, C.-S. Shih, K.-C. Li, S.-L. Tsao, Q.-A. Zeng, S.-H. Park, Y.-B. Ko, S.-H. Hung, and J. H. Park, editors, *EUC Workshops*, volume 4809 of *Lecture Notes in Computer Science*, pages 43–56. Springer, 2007.
16. P. Peris-Lopez, J. C. H. Castro, J. M. Estévez-Tapiador, and A. Ribagorda. M^2ap: A minimalist mutual-authentication protocol for low-cost rfid tags. In J. Ma, H. Jin, L. T. Yang, and J. J. P. Tsai, editors, *Ubiquitous Intelligence and Computing, Third International Conference*, volume 4159 of *Lecture Notes in Computer Science*, pages 912–923. Springer, 2006.
17. P. Peris-Lopez, J. C. H. Castro, J. M. Estévez-Tapiador, and A. Ribagorda. Advances in Ultralightweight Cryptography for Low-Cost RFID Tags: Gossamer Protocol. In K.-I. Chung, K. Sohn, and M. Yung, editors, *WISA*, volume 5379 of *LNCS*, pages 56–68. Springer, 2008.
18. P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda. EMAP: An Efficient Mutual Authentication Protocol for Low-Cost RFID Tags. In *OTM Federated Conferences and Workshop: IS Workshop - IS'06*, volume 4277 of *LNCS*, pages 352–361, Montpellier, France, November 2006. Springer-Verlag.

19. P. Peris-Lopez, J. C. Hernandez-Castro, J. E. Tapiador, and J. C. A. van der Lubbe. Cryptanalysis of an EPC class-1 generation-2 standard compliant authentication protocol. *Eng. Appl. of AI*, 24(6):1061–1069, 2011.
20. P. Peris-Lopez, T. Li, J. C. Hernandez-Castro, and J. E. Tapiador. Practical attacks on a mutual authentication scheme under the EPC class-1 generation-2 standard. *Computer Communications*, 32(7-10):1185–1193, 2009.
21. A. Sadighian and R. Jalili. FLMAP: A Fast Lightweight Mutual Authentication Protocol for RFID Systems. In *The 16th IEEE International Conference On Networks (ICON 2008)*, pages 1–6, New Delhi, India, 2008.
22. A. Sadighian and R. Jalili. AFMAP: Anonymous Forward-Secure Mutual Authentication Protocols for RFID systems. In R. Falk, W. Goudalo, E. Y. Chen, R. Savola, and M. Popescu, editors, *The Third IEEE International Conference on Emerging Security Information, Systems and Technologies(SECURWARE 2009)*, pages 31–36, Athens, Greece, 2009. IEEE Computer Society.
23. M. Safkhani, N. Bagheri, and M. Naderi. Cryptanalysis of Chen *et al.*'s RFID Access Control Protocol. Cryptology ePrint Archive, Report 2011/194, 2011. `http://eprint.iacr.org/`.
24. M. Safkhani, N. Bagheri, M. Naderi, and hamid Behnam. On the Security of Wei *et al.*'s RFID Mutual Authentication Protocol. Manuscript, 2011.
25. M. Safkhani, N. Bagheri, M. Naderi, Y. Luo, and Q. Chai. Tag Impersonation Attack on Two RFID Mutual Authentication Protocols. In *FARES*, 2011.
26. M. Safkhani, N. Bagheri, P. Peris-Lopez, M. Naderi, and J. C. Hernandez-Castro. Cryptanalysis of Cho *et al.*'s Protocol, A Hash-Based Mutual Authentication Protocol for RFID Systems. Manuscript, 2011.
27. M. Safkhani and M. Naderi. Cryptanalysis and Improvement of a Lightweight Mutual Authentication Protocol for RFID system. In 7^{th} *International ISC Conference on Information Security and Cryptology 2010(ISCISC'10)*, pages 57–59, 2010.
28. M. Safkhani, M. Naderi, and N. Bagheri. Cryptanalysis of AFMAP. *IEICE Electronics Express*, 7(17):1240–1245, 2010.
29. M. Safkhani, M. Naderi, N. Bagheri, and S. K. Sanadhya. Cryptanalysis of Some Protocols for RFID Systems. Cryptology ePrint Archive, Report 2011/061, 2011. `http://eprint.iacr.org/`.
30. M. Safkhani, M. Naderi, and H. F.Rashvand. Cryptanalysis of the Fast Lightweight Mutual Authentication Protocol (FLMAP). *International Journal of Computer & Communication Technology (IJCCT)*, 2(2,3,4):182–186, 2010.
31. J. Shen, D. Choi, S. Moh, and I. Chung. A Novel Anonymous RFID Authentication Protocol Providing Strong Privacy and Security. In *2010 International Conference on Multimedia Information Networking and Security*, 2010.
32. C. C. Tan, B. Sheng, and Q. Li. Secure and Serverless RFID Authentication and Search Protocols. *IEEE Transactions on Wireless Communications*, 7(4):1400–1407, 2008.
33. C.-H. Wei, M.-S. Hwang, and A. Y. Chin. A Mutual Authentication Protocol for RFID. *IT Professional*, 13(2):20–24, 2011.
34. S. A. Weis, S. E. Sarma, R. L. Rivest, and D. W. Engels. Security and privacy aspects of low-cost radio frequency identification systems. In D. Hutter, G. Müller, W. Stephan, and M. Ullmann, editors, *SPC*, volume 2802 of *Lecture Notes in Computer Science*, pages 201–212. Springer, 2003.
35. W. W. Y. Gu. A light-weight mutual authentication protocol for ISO 18000-6B standard RFID system . In *Proceedings of ICCTA 2009*, pages 21–25, 2009.
36. T.-C. Yeh, Y.-J. Wang, T.-C. Kuo, and S.-S. Wang. Securing RFID systems conforming to EPC class 1 generation 2 standard. *Expert Syst. Appl*, 37(12):7678–7683, 2010.
37. G. G. Yiyuan Luo, Qi Chai and X. Lai. A lightweight Stream Cipher WG-7 for RFID Encryption and Authentication. In *IEEE Globecom 2010 proceedings*, 2010.
38. Zhuzhong Qian , Ce Chen, Ilsun You, Sanglu Lu. ACSP: A novel security protocol against counting attack for UHF RFID systems. In *Elsevier, Computers and Mathematics with Applications*, volume ? of *doi:10.1016/j.camwa.2011.08.030*, page ?, 2011.