On Designing Lightweight RFID Protocols for Medical IoT

Masoumeh Safkhani[1], Ygal Bendavid[2], Samad Rostampour[2,3], Nasour Bagheri[4]

[1] Computer Engineering Department, Shahid Rajaee Teacher Training University, Tehran, Iran, Postal code: 16788-15811, Tel/fax:+98-21-22970117, Safkhani@srttu.edu

[2] University of Qubec in Montreal (UQAM), Montreal, Canada, bendavid.ygal@uqam.ca

[3] Islamic Azad University, Ahvaz, Iran, samad.rostampour@iauahvaz.ac.ir

[4] Electrical Engineering Department, Shahid Rajaee Teacher Training University, Iran, NBagheri@srttu.edu

Abstract. Recently, in IEEE Transactions on Industrial Informatics, Fan *et al.* proposed a lightweight RFID protocol which has been suggested to be employed for protecting the Medical Privacy in an IoT system. However, the protocol has trivial flaws, as it is shown recently by Aghili *et al.*, in Future Generation Computer Systems. Aghili *et al.* also proposed an improved version of the protocol, based on the similar designing paradigm, called `SecLAP`. Although the protocol's designers claimed full security against all attacks in the context, we show that the proposed protocol has serious security flaws, by presenting traceability and passive secret disclosure attacks against this protocol. More precisely, we present passive partial secret disclosure attack with the complexity of eavesdropping one session of the protocol and success probability of '1'. The disclosed parameters can be used to trace the tag/reader in any later session which compromises the tag/reader privacy. In addition, we present a passive full secret disclosure attack against `SecLAP` which can disclose $2n$-bit secret key, n-bit TID and n-bit RID with the computational complexity of $27n^7$. In addition, we show that, as it is expected, Fan *et al.*'s protocol has the worse possible security in random oracle model, where the adversary's advantage after q queries to distinguish the protocol from a random oracle is $1-2^{-q}$. We also evaluate the security of `SecLAP` in the random oracle model and show that it is as insecure as its predecessor.

Keywords: RFID, Authentication, Ultralightweight, SecLAP, Passive Attack, Random Oracle Model

1 Introduction

At any stage in the medication process, a medication error could occur. A medication error can influence appropriate prescribing, order communication, product labeling, packaging, compounding, dispensing, distribution or administration. To reduce errors throughout a medication process, a wide variety of information technologies have been applied so far, e.g. computerized prescriber order entry, robotics, automated dispensing devices such as Near Field Communication (NFC) and Radio Frequency Identification (RFID) and the electronic medication administration record such as employing Implantable Medical Devices (IMD). Although employing information technologies in a medication system can reduce various errors and improve medication process's quality, however, it has its own concerns, especially regarding patient privacy. Camara *et al.* in [10] presented a comprehensive survey on the security and privacy issues in IMD. Among the recent researches, in [24], Wu *et al.* presented a survey of access control schemes for IMD, with a focus on the security incidents, IMD threat model and the development of regulations for IMD security. In [23], Wazid *et al.* proposed NFC based authentication protocol for medicine anti-counterfeiting system in the Internet of Things (IoT) environment to check the authenticity of pharmaceutical products.

RFID is a promising wireless technology which can be employed to identify or track an object in many applications and it is a vital part of an IoT based applications, given that RFID tags are very cheap and can be attached to any object to gather the required information. In an IoT based medication system, RFID tags can be attached to objects to monitor appropriate medication process or be implanted in a patient's body to monitor the critical signals or keep a history of the received medication services, which provide health care practitioners advantages to enhance clinical practice [27]. However, those tags are very constrained and may not be able to support a standard security protocol. Hence, the privacy of tag holders, e.g. a patient, could be the main concern in RFID based applications. To address this concern, several solutions/criticisms have been proposed in the literature, e.g. [18,25,20]. A survey of these protocols is accessible through [9].

Fan *et al.* [14] recently proposed a lightweight RFID mutual authentication scheme and suggested to be employed for medical privacy protection in IoT. They have analyzed the security of the proposed protocol and claimed that it provides tag anonymity, replay attack resistance, synchronization attack resistance, forward secrecy, mutual authentication, and DoS attack resistance. However, it comes out very soon that the proposed protocol suffers from trivial weaknesses, as it is shown by Aghili and Malla [1] and more comprehensively later by Aghili *et al.* [2]. In those works, the security of the proposed scheme is scrutinized and important security pitfalls are shown. More precisely, they present an efficient secret disclosure attack and traceability attacks that violate the designers' claims. In addition, Aghili *et al.* [2] also proposed an improved protocol called `SecLAP` following the same deigning strategy and claimed optimum security against an active adversary who can control the channel between the tag and the reader and also the reader and the server.

In this paper, we evaluate the security of `SecLAP` [2], as the last member of this stream. Our analysis demonstrates that their protocol is as insecure as its predecessor and a passive adversary can disclose secret parameters that are shared between the protocol parties which is enough to mount other attacks such as traceability or desynchronization attack. Although they considered the channel between the reader and the server insecure also, we adopt the proposed attack for the case where that channel is secure and the adversary only has access to the channel between the reader and the tag and not the channel between the reader and the server.

The Paper's Contribution: This paper's contribution has the bellow folds:

1. We investigate the security of Fan *et al.*'s protocol in random oracle model (ROM) and show that the adversary's advantage to distinguish it from an ideal protocol, in which the adversary should not be able to link any two messages transfered over insecure channel together, is $1 - q^{-2}$, where q the complexity of eavesdropped messages. Although Fan *et al.*'s protocol is already known to be insecure, but this is another look at the protocol which provide better understanding the design.
2. We provide the first third party security analysis of `SecLAP`, which is the successor of Fan *et al.*'s protocol and has been very recently published in "Future Generation Computer Systems". `SecLAP` could be the latest attempt to design a secure ultra lightweight protocol for constrained environments. However, our analysis demonstrated that it is not much secure compared to Fan *et al.*'s protocol. All attack presented against this protocol are in passive adversary model. More precisely we present a passive attack which can disclose partially secret parameters of the protocol with the complexity of eavesdropping of one session of the protocol and negligible computational complexity, it can be done even by pen and paper. The extracted information are enough to trace the tag or the reader in later sessions of the protocol. In addition we present a full secret disclosure attack that can extract all secret parameters of the protocol with the complexity of $27n^7$.
3. In the discussion part of the paper, we argue that it may be better to do not attempt to design a secure ultra-lightweight protocol, because it could be simply "*Impossible Mission*".

Paper Organization : The required preliminaries and the notations that are used in this paper along with a brief description of Fan *et al.* [14] and `SecLAP` protocols are presented in Section 2. A review of the security of Fan *et al.*'s protocol is presented in Section 3. We present the result of our investigation on the security level of `SecLAP` protocol in Section 4. We discuss our suggestions to improve the protocol and conclude the paper in Section 5.

2 Preliminaries

2.1 Notations

Table 1 depicts the notations used in the rest of this paper. Fan *et al.* introduced an special bit-oriented operation in their protocol, called $Cro(X, Y)$. Given

$X = X_0, ..., X_{n-1}$ and $Y = Y_0, ..., Y_{n-1}$ as the binary representation of X and Y respectively, where X_i denotes the i^{th} bit of string X, the $Cro(X, Y) = O_0, ..., O_{2n-1}$ operation is defined as below, for $i = 0, ..., [n-1]$:

$$O_{2i} = (\bar{X} \| Y)_{2i} \oplus (\bar{Y} \| X)_{2i+1}$$

$$O_{2i+1} = (\bar{X} \| Y)_{2i+1} \oplus (\bar{Y} \| X)_{2i}$$

We provide an example to show the functionality of $Cro(X, Y)$ as follows:

X=0110 0101;

Y=1011 0111;

$\bar{X}$=1001 1010;

$\bar{Y}$=0100 1000;

$\bar{Y} \| X$=0100 1000 0110 0101;

$\bar{X} \| Y$=1001 1010 1011 0111;

$Cro(X, Y)$=0001 1110 0010 1101;

Based on this expression, the length of $Cro(X, Y)$ will be equal to the length of $\bar{X} \| Y$ and $\bar{Y} \| X$ which is twice that of X or Y. On the other hand, later we see that this function is used to update a secret value K as $K_{new} = Cro(N_R \oplus N_S \oplus N_T, K)$ which makes confusion, as the length of K_{new} then will be twice the length of K. However, following personal communication with the protocol designers [13], they stated that to reduce the length of the produced K_{new}, XOR operation is performed on the top and the bottom of the $Cro()$ operation result, see Fig. 1. In this way, the length of K_{new} and K_{old} would be the same, before the next round of authentication begins.

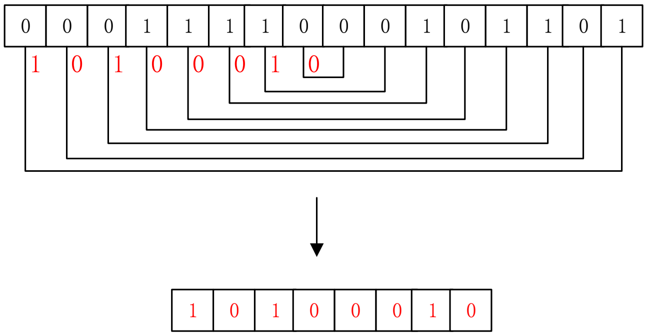


Fig. 1. To reduce the output length of $Cro(X, Y)$ [13]

To improve the Fan *et al.*'s protocol, Aghili *et al.* proposed a new component called modular rotation function $MRot_{(K)}(X, Y)$ which is used in the structure of SecLAP. Given two n-bit strings X and Y and a $2n$-bit string $K = k_1 \| k_0$, the modular rotation function $MRot_{(K)}(X, Y)$ is defined as follows:

Table 1. Notations used in this paper

Symbol	Description
R	An RFID reader
RID	The identification value of the reader R
S	A cloud server
T	An RFID tag
TID	The identification value (ID) of the tag T
K	The current session number
K_{new}	The new session number
$PRNG()$	The pseudo random number generator
$Cro(x, y)$	The bit-oriented operation defined by Fan *et al.*
$Rot(x, y)$	The rotation of sting x based on the hamming weight of y, i.e. $W(y)$
$Rot(x, y)$	The rotation of string x to left
$MRot_K(x, y)$	The bit-oriented operation defined which is used in `SecLAP`
$od(x)$	Odd bits of string x.
$ev(x)$	Even bits of string x.
$\oplus$	The bitwise XOR operation
$\Vert$	The concatenation operation
$Mark$	Two temporary bits that are used to indicate the status of the last session
$\bar{X}$	The bitwise complement of the string X

- The odd bits of X are concatenated with the even bits of Y, the result is XORed by the bits of k_1 and the result is rotated to the left depending on the value of $(Y \oplus k1)\ mod\ n$. The result is used as the odd bits of the final result. We can represent it as $[(od(X) \| ev(Y)) \oplus k_1] \lll [(Y \oplus k_0)\ mod\ n]$, where $od(X)$ and $ev(Y)$ respectively denote the odd bits of X and the even bits of Y.
- The even bits of X are concatenated with the odd bits of Y, the result is XORed by the bits of k_0 and the result is rotated to the left depending on the value of $(Y \oplus k2)\ mod\ n$. The result is used as the even bits of the final result. We can represent it as $[(ev(X) \| od(Y)) \oplus k_0] \lll [(Y \oplus k_1)\ mod\ n]$.

It is clear $MRot_{(K)}(X, Y)$ is more complicated compared to $Cro(X, Y)$ and it is also a function of the secret key K. Thanks to this property, the designers expected to provide optimum security for their protocol when they use $MRot_{(K)}(X, Y)$ as the core function and the source of confusion. It should be noted the output length of $MRot_{(K)}(X, Y)$ function, similar to $Cro(X, Y)$, will be equal to twice that of X or Y.

2.2 Random Oracle Model

A Random Oracle [8] is a theoretical black box which responses to every query with a (truly) random response chosen uniformly from its output domain. Its output to the identical inputs is the same. A Random Oracle, denoted by $\mathcal{R}$, is defined by $\mathcal{R} : \{0, 1\}^* \to \{0, 1\}^\infty$. Here $\mathcal{R}$ is a function chosen uniformly at random from the set of all functions with the same domain and range.

In general (as in our paper), the output's length of $\mathcal{R}$ is predefined to some fixed value, e.g. b bits. Hence, we re-define $\mathcal{R}$ as $\mathcal{R}_b : \{0, 1\}^* \to \{0, 1\}^b$. Given an input $x \in \{0, 1\}^*$, $\mathcal{R}_b$ will give $y = \mathcal{R}_b(x) \in \{0, 1\}^b$ as the output.

Adversary: We consider a computationally unbounded adversary with access to $\mathcal{R}_m$ or a real cryptographic component $\mathcal{C}$, which could be a cryptographic protocol $\mathcal{P}$ and m is the total length of the observable messages in each query. The adversary's "running time" is determined by the number of oracle queries that it makes to $\mathcal{R}_n/\mathcal{C}$. We use the symbol O (big-Oh), for "the expected running time of at most" and Ω (big-Omega), for "the expected running time not less than".

Indistinguishability In the cryptology terminology, $\mathcal{R}_m$ is considered as an ideal system and the distinguisher tries to distinguish the candidate crypto system $\mathcal{C}$ with m-bit output length from $\mathcal{R}_m$. In the framework of indistinguishability, the distinguisher faced with either $\mathcal{C}$ or $\mathcal{R}_m$ and aims to understand whether interacts with $\mathcal{C}$ or $\mathcal{R}_m$. Now we present the formal definition of indistinguishability following [16]:

Definition **1** *.$\mathcal{C}$ and $\mathcal{R}_m$ are (computationally) indistinguishable if for any (computationally efficient) algorithm D (called distinguisher), interacting with one of these components and generating a binary output* (0 *or* 1)*, it holds that:*

$\left|Pr\left[D^{\mathcal{C}}=1\right]-Pr\left[D^{\mathcal{R}_m}=1\right]\right|<\epsilon$

where ϵ is a negligible function of the security parameter k.

In this framework, the maximum number of queries is bounded and denoted by q.

Definition **2** *. A crypto system $\mathcal{C}$ is said to be (t_D, q, ϵ) indistinguishable from an ideal primitive $\mathcal{R}_m$ if for any distinguisher D it holds that:*

$$\left|Pr\left[D^{\mathcal{C}}=1\right]-Pr\left[D^{\mathcal{R}_m}=1\right]\right|<\epsilon$$

The distinguisher runs in time at most t_D and makes at most q queries to $\mathcal{C}/\mathcal{R}_m$. $\mathcal{C}$ is said to be (computationally) indistinguishable from $\mathcal{R}_m$ if ϵ is a negligible function of the security parameter k.

2.3 Fan *et al.*'s Protocol Description

The Fan *et al.* authentication protocol, as depicted in Fig. 2, proceeds as below:

1. The reader starts the authentication by generating a new random number N_R and sending it along with $Query$ to the tag.
2. Once the tag receives the message, it:
   - generates a random number N_T;
   - computes $Cro(RID \oplus TID, K)$;
   - sets $Mark = 00$
   - and sends N_T and $Cro(RID \oplus TID, K)$ to the reader;
3. Upon receipt of the message, the reader stores N_T and sends N_R, N_T and $Cro(RID \oplus TID, K)$ to the server.
4. When receives the message, the server:
   - stores N_R and N_T;
   - searches its database for any entry related to received $Cro(RID \oplus TID, K)$;
   - generates another random number N_S;
   - computes $Cro(RID \oplus TID, K \oplus N_S)$, $Rot(K \oplus TID, K \oplus RID) \| (N_S \oplus K)$ and sends them to the reader.
5. Once the reader receives the message, it:
   - retrieves TID and N_S from $Rot(K \oplus TID, K \oplus RID)$ and $N_S \oplus K$ respectively;
   - computes $Cro(RID \oplus TID, K \oplus N_S)$ and checks whether it equals with the received $Cro(RID \oplus TID, K \oplus N_S)$, if it does not then it stops the protocol otherwise calculates $TID \oplus N_R$;
   - and sends $TID \oplus N_R$ and N_S to the tag;
6. Once receipt of the message, the tag:
   - retrieves TID as $TID \oplus N_R \oplus N_R$;

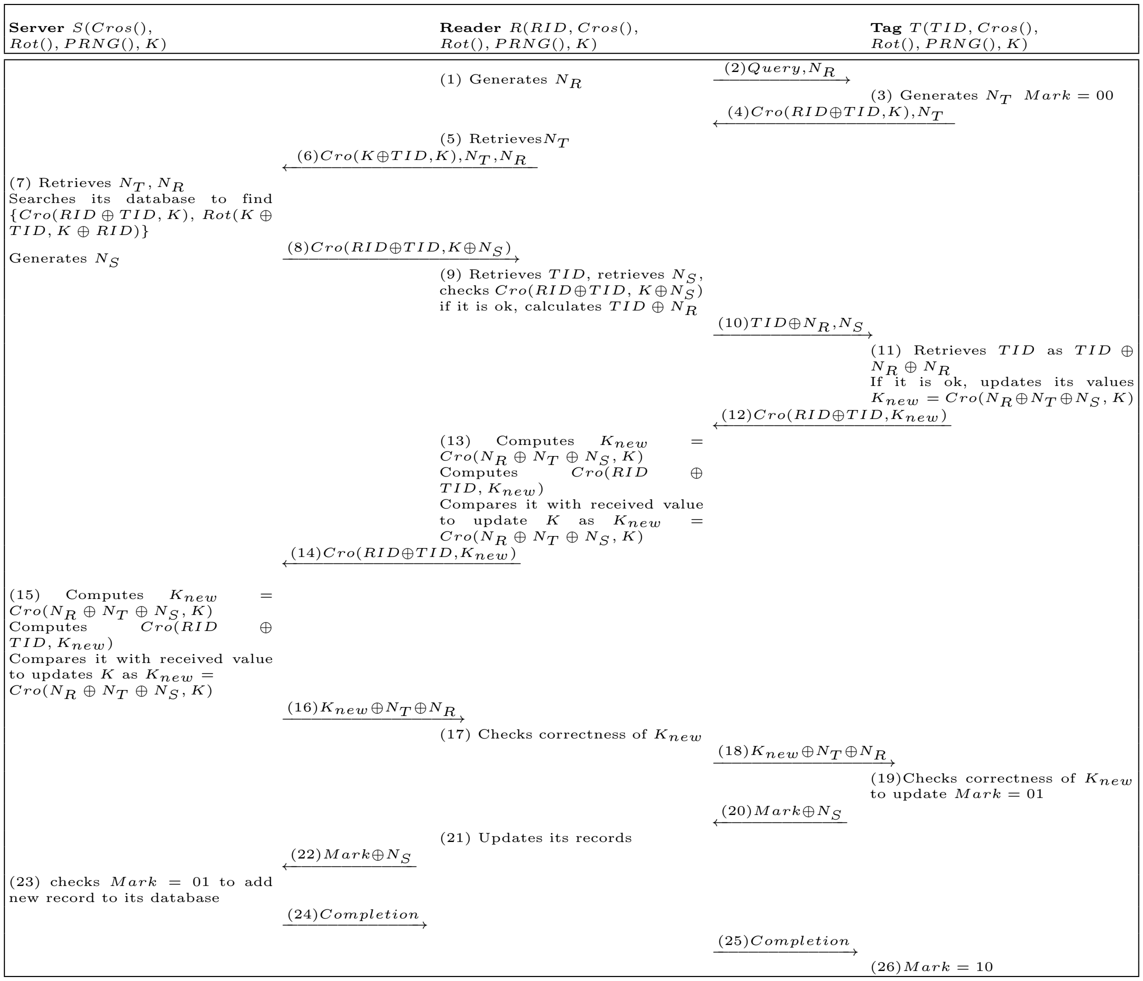
- checks whether retrieved TID equals with its TID, if it does not then it stops the protocol otherwise updates its key as $K_{new} = Cro(N_R \oplus N_S \oplus N_T, K)$;
- computes $Cro(RID \oplus TID, K_{new})$;
- and sends $Cro(RID \oplus TID, K_{new})$ to the reader.

7. When the reader receives the message, it:
   - calculates $Cro(RID \oplus TID, Cro(N_R \oplus N_S \oplus N_T, K))$ and checks whether it equals to the corresponding received value, if it does not then it terminates the protocol otherwise updates K as $K_{new} = Cro(N_R \oplus N_S \oplus N_T, K)$
   - and sends $Cro(RID \oplus TID, K_{new})$ to the server.
8. Once the server receives the message, it:
   - calculates $Cro(RID \oplus TID, Cro(N_R \oplus N_S \oplus N_T, K))$ and checks whether it equals to the corresponding received value, if it does not then it terminates the protocol otherwise updates K as $K_{new} = Cro(N_R \oplus N_S \oplus N_T, K)$;
   - and sends $K_{new} \oplus N_T \oplus N_R$ to the reader.
9. Upon receipt of the message, the reader extracts K_{new}, verifies it and if it passed the verification, sends $K_{new} \oplus N_T \oplus N_R$ to the tag.
10. The tag verifies the correctness of K_{new} by computing $K_{new} \oplus N_T \oplus N_R \oplus N_T \oplus N_R$ and if K_{new} passed the verification, tag sets $Mark = 01$ and sends $Mark \oplus N_S$ to the server through the reader.
11. The server:
    - checks whether $Mark$ equals to 01, if it is, a new record $\{Cro(RID \oplus TID, K_{new}), Rot(K_{new} \oplus TID, K_{new} \oplus RID)\}$ will be generated and added in its database;
    - sends a record completion notification to the tag via the reader.
12. When the tag received the message, sets $Mark = 10$.

2.4 Description of SecLAP

SecLAP has been proposed to improve the security drawbacks of its predecessor protocol by Fan *et al.* based on a similar designing paradigm, proceeds as below and also is depicted in Fig. 3:

1. The reader starts the authentication by generating a new random number N_R and sending it along with $Query$ to the tag.
2. Once the tag receives the message, it:
   - generates a random number N_T;
   - computes $M_1 = MRot_{(K_i)}(TID \oplus N_R, od(K_i) \oplus N_T)$ and $M_2 = MRot_{(K_i)}(ev(K_i) \oplus N_R, TID \oplus N_T)$;
   - and sends $N_T \| M_1 \| M_2$ to the reader;



Server $S(Cros(), Rot(), PRNG(), K)$

Reader $R(RID, Cros(), Rot(), PRNG(), K)$

Tag $T(TID, Cros(), Rot(), PRNG(), K)$

(1) Generates N_R

$(2) Query, N_R$

(3) Generates N_T $Mark = 00$

$(4) Cro(RID \oplus TID, K), N_T$

(5) RetrievesN_T

$(6) Cro(K \oplus TID, K), N_T, N_R$

(7) Retrieves N_T, N_R
Searches its database to find $\{Cro(RID \oplus TID, K),\ Rot(K \oplus TID, K \oplus RID)\}$

Generates N_S

$(8) Cro(RID \oplus TID, K \oplus N_S)$

(9) Retrieves TID, retrieves N_S, checks $Cro(RID \oplus TID,\ K \oplus N_S)$ if it is ok, calculates $TID \oplus N_R$

$(10) TID \oplus N_R, N_S$

(11) Retrieves TID as $TID \oplus N_R \oplus N_R$
If it is ok, updates its values $K_{new} = Cro(N_R \oplus N_T \oplus N_S, K)$

$(12) Cro(RID \oplus TID, K_{new})$

(13) Computes $K_{new} = Cro(N_R \oplus N_T \oplus N_S, K)$
Computes $Cro(RID \oplus TID, K_{new})$
Compares it with received value to update K as $K_{new} = Cro(N_R \oplus N_T \oplus N_S, K)$

$(14) Cro(RID \oplus TID, K_{new})$

(15) Computes $K_{new} = Cro(N_R \oplus N_T \oplus N_S, K)$
Computes $Cro(RID \oplus TID, K_{new})$
Compares it with received value to updates K as $K_{new} = Cro(N_R \oplus N_T \oplus N_S, K)$

$(16) K_{new} \oplus N_T \oplus N_R$

(17) Checks correctness of K_{new}

$(18) K_{new} \oplus N_T \oplus N_R$

(19)Checks correctness of K_{new} to update $Mark = 01$

$(20) Mark \oplus N_S$

(21) Updates its records

$(22) Mark \oplus N_S$

(23) checks $Mark = 01$ to add new record to its database

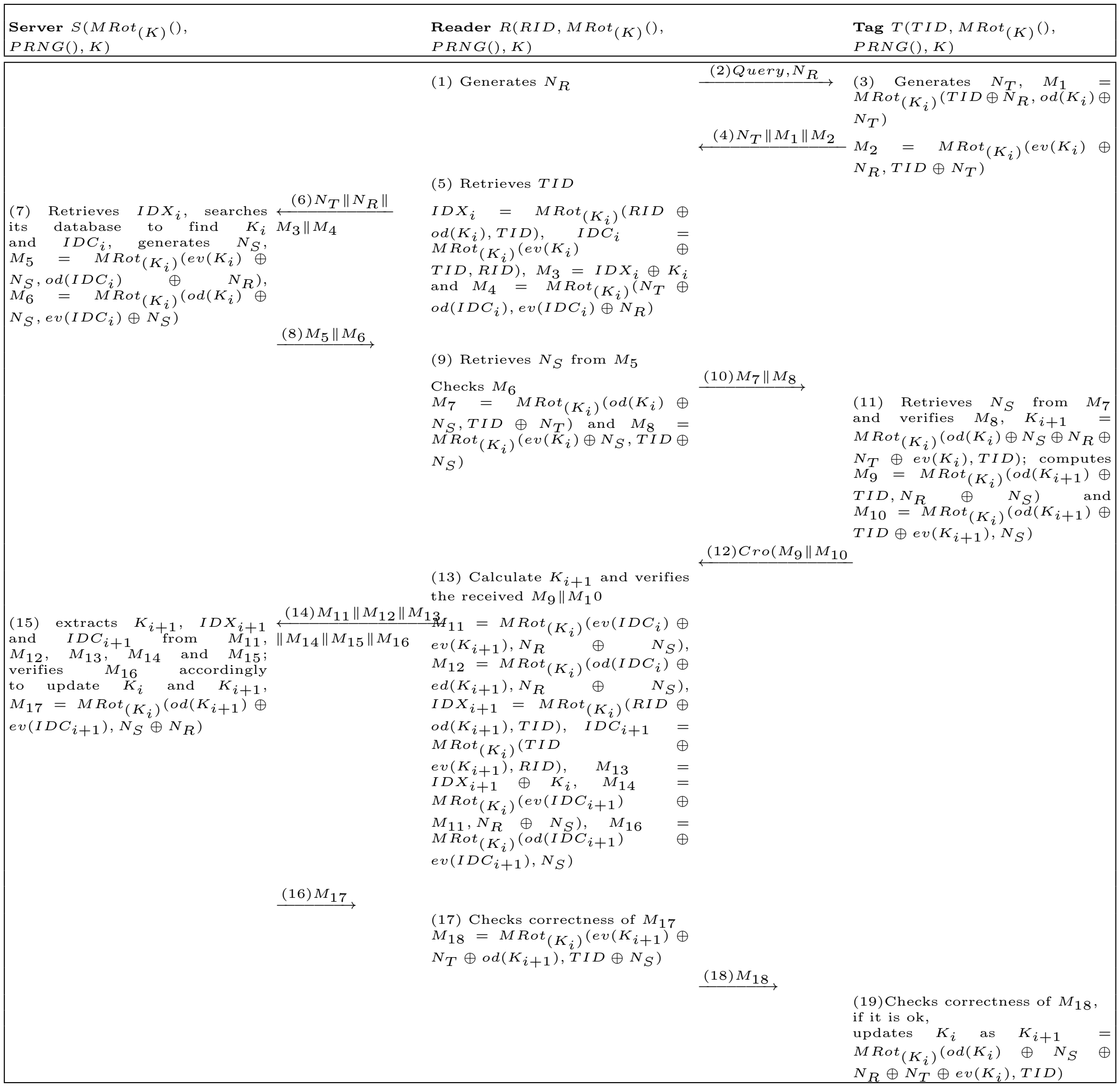
$(24) Completion$

$(25) Completion$

$(26) Mark = 10$

Fig. 2. Mutual authentication phase of Fan *et al.*'s protocol [14]

3. Upon receipt of the message, the reader obtains TID from M_1 and verifies the received M_2 based on it. Then, the reader computes $IDX_i = MRot_{(K_i)}(RID \oplus od(K_i), TID)$, $IDC_i = MRot_{(K_i)}(ev(K_i) \oplus TID, RID)$, $M_3 = IDX_i \oplus K_i$ and $M_4 = MRot_{(K_i)}(N_T \oplus od(IDC_i), ev(IDC_i) \oplus N_R)$, and forwards $N_T \| N_R \| M_3 \| M_4$ to the server. If the tag has not been authenticated the protocol is terminated.
4. When receives the message, the server:
   - extracts IDX_i and finds related K_i and IDC_i in its database IDT and verifies the received M_4;
   - generates another random number N_S;
   - computes $M_5 = MRot_{(K_i)}(ev(K_i) \oplus N_S, od(IDC_i) \oplus N_R)$, $M_6 = MRot_{(K_i)}(od(K_i) \oplus N_S, ev(IDC_i) \oplus N_S)$ and sends $M_5 \| M_6$ to the reader.
5. Once the reader receives $M_5 \| M_6$, it:
   - retrieves N_S from M_6 and verifies the received M_5 accordingly;
   - computes $M_7 = MRot_{(K_i)}(od(K_i) \oplus N_S, TID \oplus N_T)$ and $M_8 = MRot_{(K_i)}(ev(K_i) \oplus N_S, TID \oplus N_S)$;
   - and sends $M_7 \| M_8$ to the tag;
6. Once receipt the message $M_7 \| M_8$, the tag:
   - retrieves N_S from M_7 and verifies M_8 to authenticate the server/reader;
   - calculates $K_{i+1} = MRot_{(K_i)}(od(K_i) \oplus N_S \oplus N_R \oplus N_T \oplus ev(K_i), TID)$;
   - computes $M_9 = MRot_{(K_i)}(od(K_{i+1}) \oplus TID, N_R \oplus N_S)$ and $M_{10} = MRot_{(K_i)}(od(K_{i+1}) \oplus TID \oplus ev(K_{i+1}), N_S)$;
   - and sends $M_9 \| M_{10}$ to the reader.
7. When the reader receives the message, it:
   - calculates $K_{i+1} = MRot_{(K_i)}(od(K_i) \oplus N_S \oplus N_R \oplus N_T \oplus ev(K_i), TID)$ and verifies the received $M_9 \| M_{10}$;
   - computes $M_{11} = MRot_{(K_i)}(ev(IDC_i) \oplus ev(K_{i+1}), N_R \oplus N_S)$, $M_{12} = MRot_{(K_i)}(od(IDC_i) \oplus ed(K_{i+1}), N_R \oplus N_S)$, $IDX_{i+1} = MRot_{(K_i)}(RID \oplus od(K_{i+1}), TID)$, $IDC_{i+1} = MRot_{(K_i)}(TID \oplus ev(K_{i+1}), RID)$, $M_{13} = IDX_{i+1} \oplus K_i$, $M_{14} = MRot_{(K_i)}(ev(IDC_{i+1}) \oplus M_{11}, N_R \oplus N_S)$, $M_{16} = MRot_{(K_i)}(od(IDC_{i+1}) \oplus ev(IDC_{i+1}), N_S)$;
   - and sends $M_{11} \| M_{12} \| M_{13} \| M_{14} \| M_{15} \| M_{16}$ to the server.
8. Once the server receives the message, it:
   - extracts K_{i+1}, IDX_{i+1} and IDC_{i+1} respectively from M_{11}, M_{12}, M_{13}, M_{14} and M_{15} and verifies M_{16} accordingly to update K_i and K_{i+1},
   - adds $(IDX_{i+1}, K_{i+1}, IDC_{i+1})$ to its database IDT,
   - computes $M_{17} = MRot_{(K_i)}(od(K_{i+1}) \oplus ev(IDC_{i+1}), N_S \oplus N_R)$ and sends it to the reader.
9. Upon receipt of the message M_{17}, the reader verifies it to decide whether update K_i as K_{i+1}. If M_{17} has been verified correctly, the reader computes $M_{18} = MRot_{(K_i)}(ev(K_{i+1}) \oplus N_T \oplus od(K_{i+1}), TID \oplus N_S)$ and sends it to the tag.
10. The tag verifies the correctness of received M_{18} to update K_i as $K_{i+1} = MRot_{(K_i)}(od(K_i) \oplus N_S \oplus N_R \oplus N_T \oplus ev(K_i), TID)$ and the protocol is completed.



Server $S(MRot_{(K)}(), PRNG(), K)$

Reader $R(RID, MRot_{(K)}(), PRNG(), K)$

Tag $T(TID, MRot_{(K)}(), PRNG(), K)$

(1) Generates N_R

$\xrightarrow{(2) Query, N_R}$

(3) Generates N_T, $M_1 = MRot_{(K_i)}(TID \oplus N_R, od(K_i) \oplus N_T)$
$M_2 = MRot_{(K_i)}(ev(K_i) \oplus N_R, TID \oplus N_T)$

$\xleftarrow{(4) N_T \| M_1 \| M_2}$

(5) Retrieves TID
$IDX_i = MRot_{(K_i)}(RID \oplus od(K_i), TID)$, $IDC_i = MRot_{(K_i)}(ev(K_i) \oplus TID, RID)$, $M_3 = IDX_i \oplus K_i$ and $M_4 = MRot_{(K_i)}(N_T \oplus od(IDC_i), ev(IDC_i) \oplus N_R)$

$\xleftarrow{(6) N_T \| N_R \|}$ $M_3 \| M_4$

(7) Retrieves IDX_i, searches its database to find K_i and IDC_i, generates N_S, $M_5 = MRot_{(K_i)}(ev(K_i) \oplus N_S, od(IDC_i) \oplus N_R)$, $M_6 = MRot_{(K_i)}(od(K_i) \oplus N_S, ev(IDC_i) \oplus N_S)$

$\xrightarrow{(8) M_5 \| M_6}$

(9) Retrieves N_S from M_5
Checks M_6
$M_7 = MRot_{(K_i)}(od(K_i) \oplus N_S, TID \oplus N_T)$ and $M_8 = MRot_{(K_i)}(ev(K_i) \oplus N_S, TID \oplus N_S)$

$\xrightarrow{(10) M_7 \| M_8}$

(11) Retrieves N_S from M_7 and verifies M_8, $K_{i+1} = MRot_{(K_i)}(od(K_i) \oplus N_S \oplus N_R \oplus N_T \oplus ev(K_i), TID)$; computes $M_9 = MRot_{(K_i)}(od(K_{i+1}) \oplus TID, N_R \oplus N_S)$ and $M_{10} = MRot_{(K_i)}(od(K_{i+1}) \oplus TID \oplus ev(K_{i+1}), N_S)$

$\xleftarrow{(12) Cro(M_9 \| M_{10}}$

(13) Calculate K_{i+1} and verifies the received $M_9 \| M_1 0$
$M_{11} = MRot_{(K_i)}(ev(IDC_i) \oplus ev(K_{i+1}), N_R \oplus N_S)$, $M_{12} = MRot_{(K_i)}(od(IDC_i) \oplus ed(K_{i+1}), N_R \oplus N_S)$, $IDX_{i+1} = MRot_{(K_i)}(RID \oplus od(K_{i+1}), TID)$, $IDC_{i+1} = MRot_{(K_i)}(TID \oplus ev(K_{i+1}), RID)$, $M_{13} = IDX_{i+1} \oplus K_i$, $M_{14} = MRot_{(K_i)}(ev(IDC_{i+1}) \oplus M_{11}, N_R \oplus N_S)$, $M_{16} = MRot_{(K_i)}(od(IDC_{i+1}) \oplus ev(IDC_{i+1}), N_S)$

$\xleftarrow{(14) M_{11} \| M_{12} \| M_{13}}$ $\| M_{14} \| M_{15} \| M_{16}$

(15) extracts K_{i+1}, IDX_{i+1} and IDC_{i+1} from M_{11}, M_{12}, M_{13}, M_{14} and M_{15}; verifies M_{16} accordingly to update K_i and K_{i+1}, $M_{17} = MRot_{(K_i)}(od(K_{i+1}) \oplus ev(IDC_{i+1}), N_S \oplus N_R)$

$\xrightarrow{(16) M_{17}}$

(17) Checks correctness of M_{17}
$M_{18} = MRot_{(K_i)}(ev(K_{i+1}) \oplus N_T \oplus od(K_{i+1}), TID \oplus N_S)$

$\xrightarrow{(18) M_{18}}$

(19)Checks correctness of M_{18}, if it is ok,
updates K_i as $K_{i+1} = MRot_{(K_i)}(od(K_i) \oplus N_S \oplus N_R \oplus N_T \oplus ev(K_i), TID)$

Fig. 3. Mutual authentication phase of `SecLAP` [2]

3 Security Analysis of Fan *et al.*'s Protocol

3.1 Secret Disclosure and Traceability Attacks

Fan *et al.*'s protocol has trivial vulnerabilities, also reported by Aghili and Mala [1]. More precisely, consider a passive adversary which just eavesdrops the transferred messages between a legitimate reader R and the target tag T. In this case, the adversary achieves the following set of information that is transferred over an insecure channel between T and R:

- N_R sent from R to T;
- $Cro(RID \oplus TID, K)$ and N_T sent from T to R;
- $TID \oplus N_R$ and N_S sent from R to T;
- $Cro(RID \oplus TID, K_{new})$ sent from T to R;
- $K_{new} \oplus N_T \oplus N_R$ sent from R to T;
- $Mark \oplus N_S$ sent from T to R;

Given the above information, it would be trivial to extract, TID and K_{new}, as follows:

$$TID = (TID \oplus N_R) \oplus N_R;$$

$$K_{new} = (K_{new} \oplus N_T \oplus N_R) \oplus N_T \oplus N_R.$$

Given that TID is a constant parameter which determines the tag identity, it can be employed to trace the tag holder. This fact compromises the security of the protocol against traceability and tag anonymity.

Remark 1. This attack does not depend on the definition of $Cro(X, Y)$ and works for any $Cro(X, Y)$. Hence it is not possible to improve the security of the protocol against this attack by just using a more complicated function instead of the current $Cro(X, Y)$.

On the other hand, on Step 6, the tag sends $Cro(RID \oplus TID, K_{new})$ to the reader over public channel. Now assume that the adversary also eavesdrops that message between R and T, where the values of TID and K_{new} have been extracted by the adversary already.

Given $Cro(RID \oplus TID, K')$, we denote $C_i = (Cro(RID \oplus TID, K'))_i$ for simplicity, i.e. its i^{th} bit. Given that $\overline{\mathcal{X}} = \mathcal{X} \oplus 1$ any $\mathcal{X} \in \{0, 1\}$, for an even value of n, we can extract the following equations, for different cases of i:

$$\begin{cases} 0 \leq i \leq \frac{l}{2} - 1 & C_{2i} = RID_{2i+1} \oplus TID_{2i+1} \oplus K'_{2i} \\ & C_{2i+1} = RID_{2i} \oplus TID_{2i} \oplus K'_{2i+1} \\ \frac{l}{2} \leq i \leq n - 1 & C_{2i+1} = RID_{2i+l} \oplus TID_{2i+l} \oplus 1 \oplus K'_{2i} \oplus 1 \\ & C_{2i+1} = RID_{2i} \oplus TID_{2i} \oplus 1 \oplus K'_{2i+1} \oplus 1 \end{cases} \tag{1}$$

where n is the length of parameters in the protocol, e.g. $TID \in \{0, 1\}^n$. In Eq. 1, exclude RID_j for $0 \leq j \leq n-1$, all variables are known. Hence, RID will be retrieved easily from that linear equations. Given RID, the reader anonymity is compromised.

Remark 2. It should be noted the success probability of the presented attack to recover RID depends on the exact definition of $Cro(X, Y)$. Hence, it may be possible to improve the security of the protocol against this attack by just using a more complicated function instead of the current $Cro(X, Y)$.

3.2 Fan *et al.*'s Protocol Distinguishability in ROM

Any cryptographic scheme is expected to behave unpredictable form the adversary's point of view. To model this property, indistinguishability from a random oracle is used, which we defined in Section 2.2. In this section, we show that Fan *et al.*'s protocol is not a secure protocol in ROM. Given that for any value of x we can state that $x \oplus 1 \oplus 1 = x \oplus 0 = x$, Eq. 1 can be rewritten as follows:

$$\begin{cases} 0 \leq i \leq \frac{n}{2} - 1 & C_{2i} = RID_{2i+1} \oplus TID_{2i+1} \oplus K'_{2i} \\ & C_{2i+1} = RID_{2i} \oplus TID_{2i} \oplus K'_{2i+1} \\ \frac{n}{2} \leq i \leq n-1 & C_{2i+1} = RID_{2i+l} \oplus TID_{2i+l} \oplus K'_{2i} \\ & C_{2i+1} = RID_{2i} \oplus TID_{2i} \oplus K'_{2i+1} \end{cases} \tag{2}$$

It is clear that, for $i \in \{0, 2, \dots, n-2\}$ and for even values of n , $C_i = C_{n+i+1}$ and $C_{i+1} = C_{n+i}$. This property can be used as a metric to determine that the used protocol is the Fan *et al.*'s protocol and not $\mathcal{R}$. Moreover, Fan *et al.*'s protocol has zero-sum property. More precisely, it is easy to show that:

$$\oplus_{i=0}^{2n-1} C_i = C_0 \oplus C_1 \oplus \dots \oplus_{2n-1} = 0. \tag{3}$$

It comes for the fact that any variable appears in the above summation exactly two times, e.g. RID_0 or K_0, and for any $\mathcal{X} \in \{0, 1\}$ and any value of x we have $x \oplus \mathcal{X} \oplus \mathcal{X} = x \oplus 0 = x$. This property can be used as a metric to determine that the used protocol is the Fan *et al.*'s protocol or $\mathcal{R}_m$.

To distinguish Fan *et al.*'s protocol $\mathcal{P}$ from $\mathcal{R}_m$, the adversary does as follows:

1. eavesdrops a message produced by $Cro(.)$ function, transferred over channel, e.g. $C = Cro(K \oplus TID, K)$.
2. if $\oplus_{i=0}^{2n-1} C_i = 0$ returns $\mathcal{P}$; otherwise returns $\mathcal{R}_m$.

To determine the adversary's advantage, it is clear if the adversary communicates with Fan *et al.*'s protocol then with the probability of '1' returns $\mathcal{P}$ in Step 2 while if it communicates with $\mathcal{R}_m$ then returns $\mathcal{P}$ with the probability of '2^{-1}'. To improve the adversary's advantage, we can increase the number of eavesdropped messages. For an adversary who eavesdropped q messages, all produced by $Cro(.)$ function, the advantage will be as follows:

$\left| Pr\left[D^{\mathcal{C}} = 1\right] - Pr\left[D^{\mathcal{R}_m} = 1\right] \right| = 1 - 2^{-q}$ which is the maximum advantage that an adversary can get after q queries. Hence, Fan *et al.*'s protocol has the worse possible security in ROM.

4 Security Analysis of SecLAP

Similar to their predecessors, Aghili *et al.*'s also claimed optimum security of SecLAP against an active adversary, where they argued security of their protocol against different attacks informally and also formally evaluated its security using Burrows-Abadi-Needham (BAN) logic. In this section, we present the first third party evaluation of this protocol and show that, however, it is as insecure as its predecessor, i.e. Fan *et al.*'s protocol.

4.1 Partial secret disclosure attack

In this section, we present some properties of `SecLAP` that can be used to reveal information related to secret parameters. Given that information, the adversary can trace the tag or the reader which compromise their anonymity. The main observation is similar to the zero-sum property reported for Fan *et al.*'s protocol in Section 2. More precisely:

$$\oplus_{j=0}^{2n-1}(MRot_{(K_i)}(X,Y))_j = (\oplus_{j=0}^{n-1} od(MRot_{(K_i)}(X,Y))_j) \oplus (\oplus_{j=0}^{n-1} ev(MRot_{(K_i)}(X,Y))_j) \tag{4}$$

On the other hand, from the definition of $MRot_{(K_i)}(X,Y)$ we know that the $od(MRot_{(K_i)}(X,Y)) = [(od(X) \| ev(Y)) \oplus k_1] \lll [(Y \oplus k_0)\ mod\ n]$ and $ev(MRot_{(K_i)}(X,Y)) = [(ev(X) \| od(Y)) \oplus k_0] \lll [(Y \oplus k_1)\ mod\ n]$. Hence:

$$\oplus_{j=0}^{2n-1}(MRot_{(K_i)}(X,Y))_j = (\oplus_{j=0}^{n-1}[(od(X) \| ev(Y)) \oplus k_1]_j) \oplus (\oplus_{j=0}^{n-1}[(ev(X) \| od(Y)) \oplus k_0]_j) \tag{5}$$

The above equation can be simplified as follow:

$$\oplus_{j=0}^{2n-1}\ (MRot_{(K_i)}(X,Y))_j = (\oplus_{j=0}^{n-1} X_j) \oplus (\oplus_{j=0}^{n-1} Y_j) \oplus (\oplus_{j=0}^{2n-1}(K_i)_j) \tag{6}$$

Next, assume that the adversary eavesdropped the first run of the exchanged messages between the reader and the tag, i.e., N_R and $N_T \| M_1 \| M_2$, where $M_1 = MRot_{(K_i)}(TID \oplus N_R, od(K_i) \oplus N_T)$ and $M_2 = MRot_{(K_i)}(ev(K_i) \oplus N_R, TID \oplus N_T)$. Following the above argument, we can write:

$$\oplus_{j=0}^{2n-1}(M_1)_j = (\oplus_{j=0}^{n-1}(TID \oplus N_R)_j) \oplus (\oplus_{j=0}^{n-1}(od(K_i) \oplus N_T)j) \oplus (\oplus_{j=0}^{2n-1}(K_i)_j) \tag{7}$$

and:

$$\oplus_{j=0}^{2n-1}(M_1)_j \oplus (\oplus_{j=0}^{n-1}(N_R)_j) \oplus (\oplus_{j=0}^{n-1}(N_T)j) = (\oplus_{j=0}^{n-1}(TID)_j) \oplus (\oplus_{j=0}^{n-1}(ev(K_i))_j) \tag{8}$$

Similarly, we can argue that:

$$\oplus_{j=0}^{2n-1}(M_2)_j \oplus (\oplus_{j=0}^{n-1}(N_R)_j) \oplus (\oplus_{j=0}^{n-1}(N_T)j) = (\oplus_{j=0}^{n-1}(TID)_j) \oplus (\oplus_{j=0}^{n-1}(od(K_i))_j) \tag{9}$$

and, given Equations 8 and 9:

$$(\oplus_{j=0}^{2n-1}(M_1)_j) \oplus (\oplus_{j=0}^{2n-1}(M_2)_j) = \oplus_{j=0}^{2n-1}(K_i)_j \tag{10}$$

Given that the secret parameters of `SecLAP` are TID, RID and K_i, Equations 8 and 9 reveal two bits of the secrets. Given that as long as the tag has not updated its secrets those bits remain fixed, the adversary can use it as a source of traceability, which compromises the designers claim on the security of the protocol against traceability.

Given that the channel between the reader and the server is also insecure, to trace the reader, the adversary eavesdrops IDX_i, a sent message from the reader to the server, where $IDX_i = MRot_{(K_i)}(RID \oplus od(K_i), TID)$. Recall from Equation 6:

$$\oplus_{j=0}^{2n-1} (IDX_i)_j = (\oplus_{j=0}^{n-1}(RID)_j) \oplus (\oplus_{j=0}^{n-1}(TID)_j) \oplus (\oplus_{j=0}^{n-1}(ev(K_i))_j) \quad (11)$$

Combining Equations 11 and 8, reveals a single bit of RID as follows:

$$(\oplus_{j=0}^{2n-1}(IDX_i)_j) \oplus (\oplus_{j=0}^{2n-1}(M_2)_j) = \oplus_{j=0}^{n-1}(RID)_j \quad (12)$$

It worth noting the related information in Equation 12 is independent of the tag's data and also constant. Hence it is enough to compromise the reader anonymity.

Other messages that can be eavesdropped by the adversary are as follows, that can be used to disclose other information:

1. $IDC_i = MRot_{(K_i)}(ev(K_i) \oplus TID, RID)$, $M_3 = IDX_i \oplus K_i$ and $M_4 = MRot_{(K_i)}(N_T \oplus od(IDC_i), ev(IDC_i) \oplus N_R)$ sent from the reader to the server,
2. $M_5 = MRot_{(K_i)}(ev(K_i) \oplus N_S, od(IDC_i) \oplus N_R)$, $M_6 = MRot_{(K_i)}(od(K_i) \oplus N_S, ev(IDC_i) \oplus N_S)$ sent from the server to the reader,
3. $M_7 = MRot_{(K_i)}(od(K_i) \oplus N_S, TID \oplus N_T)$ and $M_8 = MRot_{(K_i)}(ev(K_i) \oplus N_S, TID \oplus N_S)$ sent from the reader to the tag,
4. $M_9 = MRot_{(K_i)}(od(K_{i+1}) \oplus TID, N_R \oplus N_S)$ and $M_{10} = MRot_{(K_i)}(od(K_{i+1}) \oplus TID \oplus ev(K_{i+1}), N_S)$ sent from the tag to the reader,
5. $M_{11} = MRot_{(K_i)}(ev(IDC_i) \oplus ev(K_{i+1}), N_R \oplus N_S)$, $M_{12} = MRot_{(K_i)}(od(IDC_i) \oplus ed(K_{i+1}), N_R \oplus N_S)$, $IDX_{i+1} = MRot_{(K_i)}(RID \oplus od(K_{i+1}), TID)$, $IDC_{i+1} = MRot_{(K_i)}(TID \oplus ev(K_{i+1}), RID)$, $M_{13} = IDX_{i+1} \oplus K_i$, $M_{14} = MRot_{(K_i)}(ev(IDC_{i+1}) \oplus M_{11}, N_R \oplus N_S)$, $M_{16} = MRot_{(K_i)}(od(IDC_{i+1}) \oplus ev(IDC_{i+1}), N_S)$ sent from the reader to the server,
6. $M_{17} = MRot_{(K_i)}(od(K_{i+1}) \oplus ev(IDC_{i+1}), N_S \oplus N_R)$ sent from the server to the reader,
7. $M_{18} = MRot_{(K_i)}(ev(K_{i+1}) \oplus N_T \oplus od(K_{i+1}), TID \oplus N_S)$ sent from the reader to the tag,

For example:

$$(\oplus_{j=0}^{2n-1}(M_6)_j) \oplus (\oplus_{j=0}^{2n-1}(IDC_i)_j) = (\oplus_{j=0}^{n-1}(ev(K_i))_j) \quad (13)$$

Now combining Equations 8 and 13, reveals a bit of TID, as follows:

$$(\oplus_{j=0}^{2n-1}(M_6)_j) \oplus (\oplus_{j=0}^{2n-1}(IDC_i)_j) \oplus (\oplus_{j=0}^{2n-1}(M_1)_j) = (\oplus_{j=0}^{n-1}(ev(K_i))_j) \oplus (\oplus_{j=0}^{n-1}(TID)_j) \oplus (\oplus_{j=0}^{n-1}(ev(K_i))_j) = \oplus_{j=0}^{n-1}(TID)_j \quad (14)$$

Given that TID will not be updated at the end of the session, it can be used to compromise the tag's anonymity and trace the tag holder in any session.

Similarly,

$$(\oplus_{j=0}^{2n-1}(M_9)_j) \oplus (\oplus_{j=0}^{2n-1}(M_{10})_j) \oplus (\oplus_{j=0}^{2n-1}(N_R)_j) = (\oplus_{j=0}^{n-1}(ev(K_{i+1}))_j) \quad (15)$$

and from IDX_{i+1} and Equation 10, 12 and 14 we can disclose $(\oplus_{j=0}^{n-1}(od(K_{i+1}))_j)$, which combined with Equation 15 reveals two bits of K_{i+1} and also $(\oplus_{j=0}^{n-1}(K_{i+1})_j)$.

The success probability of disclosing all values mentioned in this section is '1' and the complexity is only eavesdropping one session of the protocol.

4.2 Full secret disclosure attack

The secret parameters of SecLAP are $K_i \in \{0,1\}^{2n}$, $TID \in \{0,1\}^n$ and $RID \in \{0,1\}^n$, where TID and RID are constant values while K_i is a dynamic value which is updated after each successful run of the protocol. Most of the transferred messages are masked by K_i and produced by $MRot_{(K_i)}(.)$, which is more complicated than $Cro(.)$ which has been used by Fan *et al.* Hence, designers of SecLAP expect a good security of the protocol against secret disclosure attack. However, we present an attack to extract those secret parameters efficiently. Despite of the designers claim that SecLAP is secure even against an active adversary who has full control over the channel between the tag and the reader and the reader and the server, we consider the weakest adversary who can just eavesdrop the channel between the tag and the reader and even not the server and the reader. To start the attack, we assume that the adversary eavesdrops any transferred message form the tag T to the reader R or from the R to T in session i, which we call this phase of attack the **learning phase of attack**. Hence, at the end of the learning phase of the attack, the adversary has the below information:

- N_R sent from R to T;
- $M_1 = MRot_{(K_i)}(TID \oplus N_R, od(K_i) \oplus N_T)$ and $M_2 = MRot_{(K_i)}(ev(K_i) \oplus N_R, TID \oplus N_T)$ and N_T sent from T to R;
- $M_7 = MRot_{(K_i)}(od(K_i) \oplus N_S, TID \oplus N_T)$ and $M_8 = MRot_{(K_i)}(ev(K_i) \oplus N_S, TID \oplus N_S)$ sent from R to T;
- $M_9 = MRot_{(K_i)}(od(K_{i+1}) \oplus TID, N_R \oplus N_S)$ and $M_{10} = MRot_{(K_i)}(od(K_{i+1}) \oplus TID \oplus ev(K_{i+1}), N_S)$ sent from T to R;
- $M_{18} = MRot_{(K_i)}(ev(K_{i+1}) \oplus N_T \oplus od(K_{i+1}), TID \oplus N_S)$ sent from R to T;

On the other hand, from the definition of $MRot_{(K_i)}(X,Y)$ we know that the $od(MRot_{(K_i)}(X,Y)) = [(od(X)\|ev(Y)) \oplus k_1] \lll [(Y \oplus k_0)\ mod\ n]$ while $ev(MRot_{(K_i)}(X,Y)) = [(ev(X)\|od(Y)) \oplus k_0] \lll [(Y \oplus k_1)\ mod\ n]$ is used to produce the even bits of $MRot_{(K_i)}(X,Y)$. Hence, we can argue that any bit r of the output, $O_r = (MRot_{(K_i)}(X,Y)_r)$, is a linear function of a bit from K with either a bit from X or a bit from Y, i.e. $O_r = c \times (X)_s \oplus \bar{c} \times (Y)_t \oplus (K_b)_u$, where $b, c \in \{0,1\}$, $r \in \{0,1,\ldots,2n-1\}$, and $s,t,u \in \{0,1,\ldots,n-1\}$. Although we may not know the exact value of s,t and u. However, again from the definition of $MRot_{(K_i)}(X,Y)$ we know that if r is an odd value then s will also be an odd value while t will be even and if r is an even value then s will also be even while t will be odd. In addition, if $O_r = (X)_s \oplus (Y)_t \oplus (K_b)_u$ then $O_{r+2z} = (X)_{s+2z} \oplus (Y)_{t+2z} \oplus (K_b)_{u+2z}$, for any $z \in \{0,1,\ldots,n-1\}$, obviously addition takes place module n. In addition, given u and b in $(k_b)_u$, we can uniquely determine related values of s and t. It comes from the fact that $od(MRot_{(K_i)}(X,Y)) \ggg [(Y \oplus k_0)\ mod\ n] = [(od(X)\|ev(Y)) \oplus k_1]$ while $ev(MRot_{(K_i)}(X,Y)) \ggg [(Y \oplus k_1)\ mod\ n] = [(ev(X)\|od(Y)) \oplus k_0]$. Therefor, given to extract $2n$-linear equation out of $MRot_{(K_i)}(X,Y)$, the only unknown parameter will be $[(Y \oplus k_1)\ mod\ n]$ and $[(Y \oplus k_0)\ mod\ n]$, which has the total complexity of $n+n=2n$. In the rest of the paper we use odd-offset of and even-offset ef to denote $[(Y \oplus k_1)\ mod\ n]$ and $[(Y \oplus k_0)\ mod\ n]$ respectively. Now, given the eavesdropped M_1, M_2, N_R and N_T from a session of the protocol and the above properties of $MRot_{(K_i)}(X,Y)$, the adversary uses the below procedure to extract the secret parameters of `SecLAP`:

1. for $of1 = 0,...,n-1$:
2. for $ef1 = 0,...,n-1$:
   - (a) $od(M_1) \ggg of1 \rightarrow \{[od(TID \oplus N_R)\|ev(od(K_i) \oplus N_T)] \oplus k_0\}$;
   - (b) $ev(M_1) \ggg ef1 \rightarrow \{[ev(TID \oplus N_R)\|od(od(K_i) \oplus N_T)] \oplus k_1\}$;
   - (c) for $of2 = 0,...,n-1$:
   - (d) for $ef2 = 0,...,n-1$:
     - i. $od(M_2) \ggg of2 \rightarrow \{[od(ev(K_i) \oplus N_R)\|ev(TID \oplus N_T)] \oplus k_0\}$;
     - ii. $ev(M_2) \ggg ef2 \rightarrow \{[ev(ev(K_i) \oplus N_R)\|od(TID \oplus N_T)] \oplus k_1\}$;
     - iii. Steps 2a, 2b, 2(d)i and 2(d)ii produce $4n$ linear equations while the unknown parameters are $TID \in \{0,1\}^n$ and $K_i \in \{0,1\}^{2n}$. Hence, the adversary uses $3n$ linearly independent equations to determine TID and K_i and the rest of equations to filter wrong guesses.
     - iv. return the candidate TID and K_i $of1$, $ef1, of2$ and $ef2$.
3. Given that $of1 = [((od(K_i) \oplus N_T) \oplus k_1)\ mod\ n]$, $ef1 = [((od(K_i) \oplus N_T) \oplus k_0)\ mod\ n]$, $of2 = [((TID \oplus N_T) \oplus k_1)\ mod\ n]$ and $ef2 = [((TID \oplus N_T) \oplus k_0)\ mod\ n]$ the returned TID and K_i should also pass.

The above attack has the complexity of $4n$ time solving a linear equation with $4n$ equations of $3n$ independent variable. Hence, it is expected to do not return any candidate exclude the correct pair of TID and K_i in Step 2(d)iv. However, any possible wrong guess also filtered in Step 3. Given that any wrong guess passes Step 3 with the probability of n^{-4}, the algorithm will return the correct value of TID and K_i. Next, given TID and K_i and also the eavesdropped M_7 the adversary extracts N_S and calculates K_{i+1} as $K_{i+1} = MRot_{(K_i)}(od(K_i) \oplus N_S \oplus N_R \oplus N_T \oplus ev(K_i), TID)$. The adversary can also use other eavesdropped messages, i.e. M_8, M_9, M_{10} and M_{18} to filter any possible wrong guess.

Given that the computational complexity of solving a system of linear equation of N variables has the complexity of $O(N^3)$, the expected complexity of extracting TID, K_i and K_{i+1} is $n^4(3n)^3 = 27n^7$. For $n = 128 = 2^7$, the adversary will be able to extract those secret parameters with the complexity of $O(2^{54})$, while the expected complexity is at least 2^{128}, which shows a huge gape.

It should be noted, given that the channel between the reader and the server is insecure also, the adversary can eavesdrop $IDX_i = MRot_{(K_i)}(RID \oplus od(K_i), TID)$. Assuming that the adversary has already disclosed TID and K_i, it can easily extract RID also. Similarly, the adversary can use M_7 to extract N_s which, along with other known parameters, can be used to construct $K_{i+1} = MRot_{(K_i)}(od(K_i) \oplus N_S \oplus N_R \oplus N_T \oplus ev(K_i), TID)$. In this way, the adversary revealed whole secret parameters of the protocol with the computation complexity of $27n^7$ and just eavesdropping one session of the protocol between the protocol parties, i.e. tag, reader and server.

4.3 Traceability attack

In Section 4.2, we presented an attack where any passive adversary who eavesdrops the transferred messages of a session of the protocol between the legitimate tag and the reader and server will be able to extract TID, RID, K_i and K_{i+1} with the complexity of $O(2^{54})$. Given that TID and RID are constant values for any tag/reader and will not be updated after the protocol completion, they can be used to trace the tag/reader, which contradicts the designers claim.

4.4 Distinguishability in ROM

In this section, similar to the case of Fan *et al.*, we show that `SecLAP` is not also a secure protocol in ROM. To distinguish `SecLAP` from ROM we use the observation used to partial recover secret parameters in Section 4.1. From the structure of the messages one can deduce that:

$$(\oplus_{j=0}^{2n-1}(M_1)_j) \oplus (\oplus_{j=0}^{2n-1}(M_2)_j) \oplus (\oplus_{j=0}^{2n-1}(M_3)_j) \oplus (\oplus_{j=0}^{2n-1}(IDX_i)_j) = 0 \quad (16)$$

This property can be used as a metric to determine that the used protocol is `SecLAP` and not $\mathcal{R}$.

To distinguish `SecLAP` $\mathcal{P}$ from $\mathcal{R}_m$, the adversary does as follows:

1. eavesdrops messages of a session of the given protocol, transferred over channel.
2. evaluates Equation 16 and returns $\mathcal{P}$ if it is true; otherwise returns $\mathcal{R}_m$.

To determine the adversary's advantage, it is clear if the adversary communicates with `SecLAP` then with the probability of '1' returns $\mathcal{P}$ in Step 2 while if it communicates with $\mathcal{R}_m$ then returns $\mathcal{P}$ with the probability of '2^{-1}'. To improve the adversary's advantage, we can increase the number of eavesdropped sessions. For an adversary who eavesdropped q sessions, the advantage will be as follows:

$\left|Pr\left[D^{\mathcal{C}} = 1\right] - Pr\left[D^{\mathcal{R}_m} = 1\right]\right| = 1 - 2^{-q}$ which is the maximum advantage that an adversary can get after q queries. It should be noted other combination of transferred messages can also be used for distinguishing `SecLAP` from ROM. For example:

$$(\oplus_{j=0}^{2n-1}(M_1)_j)\oplus(\oplus_{j=0}^{2n-1}(M_2)_j)\oplus(\oplus_{j=0}^{2n-1}(M_7)_j)\oplus(\oplus_{j=0}^{2n-1}(M_8)_j)\oplus(\oplus_{j=0}^{2n-1}(N_T)_j) = 0 \quad (17)$$

An interesting point with Equation 17 is the fact to distinguish the protocol the adversary only requires to eavesdrop the channel between the tag and the reader. It is clear that the adversary can use both Equation 17 and 16 (and several other combinations) to achieve same advantage with eavesdropping less number of sessions.

Therefore, similar to Fan *et al.*'s protocol, SecLAP has the worse possible security in ROM.

5 Discussion and Conclusion

In the previous sections, we showed that the Fan *et al.*'s protocol and its successor SecLAP do not provide the desired security against various attacks. More interestingly, all presented attacks work in a passive adversary model and have very low data/time complexity and the success probability of 1, exclude the full secret disclosure attack against SecLAP which has the complexity of $27n^7$ which also is very low compared to the secret key length which is $2n$-bit. Although these protocols have trivial flaws and they could be designed more sophisticated, such that will not be compromised that badly, however, previous studies show that it may not be possible to design a secure authentication protocol just using several calls to ultra-lightweight operations such as the $Cro(.)$ function which has been designed by Fan *et al.* and $MRot_{(K_i)}(.)$ function which has been designed by Aghili *et al.*. It should be noted, designing a protocol from this point of view has a long unsuccessful history. SASI [11], RAPP [22], SLAP [15] and LMAP [17] are just examples that have been compromised by the following analysis [4,3,19,21]. Hence, attempting to improve the current protocols by keeping their basic structure and just improving the way that messages are calculated may not be possible and would be just proposing another easy to break protocol. Hence, we avoid designing the improved version of this protocol in this paper. However, our suggestion is to consider the recent advances in lightweight cryptography in designing new protocols. For example, many lightweight block ciphers have been proposed in recent years that can be implemented even in passive RFID tags, based on their reported implementation results. SKINNY [7], SIMON [6], SIMECK [26] and Midori [5] are just examples. In addition, the current ongoing NIST competition for lightweight cryptography [12] also announced its first-round candidates, which aims to provide secure building blocks for constrained environments such as RFID and IoT. Hence as future work, we suggest to design a secure protocol for mutual authentication based on this schemes rather than developing a new ultra-lightweight function, such as Fan *et al.*'s $Cro(.)$ function or SecLAP's $MRot_{(K_i)}(.)$ function, and designing an easily breakable protocol based on it.

It should be noted, the Fan *et al.*'s protocol and also its successor SecLAP are not efficient protocols in the term of the number of the messages transferred between the tag and the reader, which are 7 and 5 runs respectively. It should be possible to design a protocol with much less number of messages that achieves all security targets of their protocol. This can be also considered as a target for future works.

References

1. S. F. Aghili and H. Mala. Security analysis of fan et al. lightweight rfid authentication protocol for privacy protection in iot. Cryptology ePrint Archive, Report 2018/388, 2018. `https://eprint.iacr.org/2018/388`.
2. S. F. Aghili, H. Mala, P. Kaliyar, and M. Conti. Seclap: Secure and lightweight rfid authentication protocol for medical iot. *Future Generation Computer Systems*, 101:621 – 634, 2019.
3. Z. Ahmadian, M. Salmasizadeh, and M. R. Aref. Desynchronization attack on RAPP ultralightweight authentication protocol. *Inf. Process. Lett.*, 113(7):205–209, 2013.
4. G. Avoine, X. Carpent, and B. Martin. Privacy-friendly synchronized ultralightweight authentication protocols in the storm. *J. Network and Computer Applications*, 35(2):826–843, 2012.
5. S. Banik, A. Bogdanov, T. Isobe, K. Shibutani, H. Hiwatari, T. Akishita, and F. Regazzoni. Midori: A block cipher for low energy. In T. Iwata and J. H. Cheon, editors, *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II*, volume 9453 of *Lecture Notes in Computer Science*, pages 411–436. Springer, 2015.
6. R. Beaulieu, D. Shors, J. Smith, S. Treatman-Clark, B. Weeks, and L. Wingers. The SIMON and SPECK lightweight block ciphers. In *Proceedings of the 52nd Annual Design Automation Conference, San Francisco, CA, USA, June 7-11, 2015*, page 175. ACM, 2015.
7. C. Beierle, J. Jean, S. Kölbl, G. Leander, A. Moradi, T. Peyrin, Y. Sasaki, P. Sasdrich, and S. M. Sim. The SKINNY family of block ciphers and its low-latency variant MANTIS. In M. Robshaw and J. Katz, editors, *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*, volume 9815 of *Lecture Notes in Computer Science*, pages 123–153. Springer, 2016.
8. M. Bellare and P. Rogaway. Random Oracles Are Practical : A Paradigm for Designing Efficient Protocols. In V. Ashby, editor, *1st Conference on Computing and Communications Security*, pages 62–73. ACM Press, 1993.
9. M. Benssalah, M. Djeddou, and K. Drouiche. Security analysis and enhancement of the most recent RFID authentication protocol for telecare medicine information system. *Wireless Personal Communications*, 96(4):6221–6238, 2017.
10. C. Camara, P. Isasi, K. Warwick, V. Ruiz, T. Z. Aziz, J. F. Stein, and E. Bakstein. Resting tremor classification and detection in parkinson's disease patients. *Biomed. Signal Proc. and Control*, 16:88–97, 2015.
11. H.-Y. Chien. SASI: A new ultralightweight RFID authentication protocol providing strong authentication and strong integrity. *IEEE Trans. Dependable Sec. Comput.*, 4(4):337–340, 2007.
12. N. computer security resource center (csrc). Lightweight cryptography: Round 1 candidates. NIST, 2019. `https://csrc.nist.gov/projects/lightweight-cryptography/round-1-candidates`.

13. K. Fan. Clarification of $cro(x, y)$. Personal communication, 8 may 2018.
14. K. Fan, W. Jiang, H. Li, and Y. Yang. Lightweight RFID protocol for medical privacy protection in iot. *IEEE Trans. Industrial Informatics*, 14(4):1656–1665, 2018.
15. H. Luo, G. Wen, J. Su, and Z. Huang. SLAP: Succinct and lightweight authentication protocol for low-cost RFID system. *Wireless Networks*, pages 1–10, 2016.
16. U. M. Maurer, R. Renner, and C. Holenstein. Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology. In M. Naor, editor, *First Theory of Cryptography Conference, TCC*, volume 2951 of *Lecture Notes in Computer Science*, pages 21–39. Springer, 2004.
17. P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda. LMAP: A real lightweight mutual authentication protocol for low-cost RFID tags. In *Proceedings of RFIDSec06 Workshop on RFID Security*, Graz,Austria , 12-14 July 2006.
18. P. Peris-Lopeza, A. Orfila, A. Mitrokotsa, and J. C. van der Lubbe. A comprehensive RFID solution to enhance inpatient medication safety. *international journal of medical informatics*, 80:13–24, 2011.
19. R. C.-W. Phan. Cryptanalysis of a new ultralightweight RFID authentication protocol - SASI. *IEEE Transactions on Dependable and Secure Computing*, 6(4):316–320, 2009.
20. P. Picazo-Sanchez, N. Bagheri, P. Peris-Lopez, and J. E. Tapiador. Two RFID standard-based security protocols for healthcare environments. *J. Medical Systems*, 37(5):9962, 2013.
21. M. Safkhani and N. Bagheri. Generalized desynchronization attack on UMAP: application to rcia, kmap, SLAP and sasi^{+} protocols. *IACR Cryptology ePrint Archive*, 2016:905, 2016.
22. Y. Tian, G. Chen, and J. Li. A new ultralightweight RFID authentication protocol with permutation. *IEEE Communications Letters*, 16(5):702–705, 2012.
23. M. Wazid, A. K. Das, M. K. Khan, A. A. Al-Ghaiheb, N. Kumar, and A. V. Vasilakos. Secure authentication scheme for medicine anti-counterfeiting system in iot environment. *IEEE Internet of Things Journal*, 4(5):1634–1646, 2017.
24. L. Wu, X. Du, M. Guizani, and A. Mohamed. Access control schemes for implantable medical devices: A survey. *IEEE Internet of Things Journal*, 4(5):1272–1283, 2017.
25. S. Wu, K. Chen, and Y. Zhu. A secure lightweight rfid binding proof protocol for medication errors and patient safety. *Journal of Medical Systems*, pages 1–7. 10.1007/s10916-011-9750-x.
26. G. Yang, B. Zhu, V. Suder, M. D. Aagaard, and G. Gong. The simeck family of lightweight block ciphers, 2015.
27. W. Yao, C. Chu, and Z. Li. The use of rfid in healthcare: Benefits and barriers. In *RFID-Technology and Applications (RFID-TA), 2010 IEEE International Conference on*, pages 128–134, 2010.