Breaking Anonymity of Some Recent Lightweight RFID Authentication Protocols

Karim Baghery · Behzad Abdolmaleki · Shahram Khazaei · Mohammad Reza Aref

Abstract Due to their impressive advantages, Radio Frequency IDentification (RFID) systems are ubiquitously found in various novel applications. These applications are usually in need of quick and accurate authentication or identification. In many cases, it has been shown that if such systems are not properly designed, an adversary can cause security and privacy concerns for end-users. In order to deal with these concerns, impressive endeavors have been made which have resulted in various RFID authentications being proposed. In this study, we analyze three lightweight RFID authentication protocols proposed in Wireless Personal Communications (2014), Computers & Security (2015) and Wireless Networks (2016). We show that none of the studied protocols provides the desired security and privacy required by the end-users. We present various security and privacy attacks such as secret parameter reveal, impersonation, DoS, traceability, and forward traceability against the studied protocols. Our attacks are mounted in the *Ouafi-Phan* RFID formal privacy model which is a modified version of well-known *Juels-Weis* privacy model.

Keywords Anonymous RFID Authentication Protocol · Internet of Things (IoT) · Security and Privacy · Hash Functions · Ouafi-Phan Privacy Model

1 Introduction

It is indisputable that these days Radio Frequency Identification (RFID) technology and its by-products such as RFID tags and RFID smart cards play a salient role in almost every aspect of our lives that needs quick and accurate identification or authentication [1]. In fact, impressive advantages of these systems, such as low cost, accuracy, speed, user-friendliness and scalability have let them appear in lots of novel and sensitive applications such as health-care, e-passport, supply chain, Internet of Things (IoT), access control, shopping and etc [2,3,4,5]. In all these applications, an RFID tag attached to a target object is intended to be identified, tracked or authenticated in different locations and situations. Each RFID tag has a unique identification number such as an Electronic Product Code (EPC) which is registered in the database of a central computer, referred to as *server* or *back-end server*. The server has the unique identifier of each and every RFID tag. Each tag uses its own specific identifier code when communicating with the server. In some sensitive applications, beside the mentioned unique identifier, each tag has some updatable and synchronized secret keys which are shared with the server; these keys are usually used in applications that require a secure and confidential identification or authentication [6]. Basically, in an RFID system the server has unlimited power and resources,

* This is a preprint of an article published in journal of Wireless Networks, 2019, 25: 1235. The final authenticated version is available online at `https://doi.org/10.1007/s11276-018-1717-0`

K. Baghery
Institute of Computer Science, University of Tartu, Estonia.
E-mail: karim.baghery@ut.ee

B. Abdolmaleki
Institute of Computer Science, University of Tartu, Estonia.
E-mail: behzad.abdolmaleki@ut.ee

S. Khazaei
Department of Mathematical Sciences, Sharif University of Technology, Tehran, Iran.
E-mail: shahram.khazaei@sharif.edu

M. R. Aref
ISSL Lab, Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran.
E-mail: aref@sharif.edu

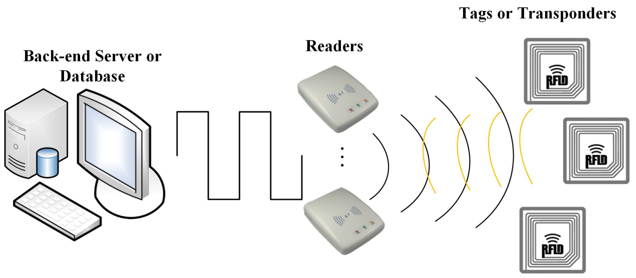


Fig. 1: A typical RFID system model

whereas the tag has limited power and capabilities. Due to this fact, the majority of identification or authentication process is carried out in the server side and the tag process should be lightweight [7].

In general, an RFID system consists of a large number of tags, some readers and a back-end server. A graphical illustration of a typical RFID system is shown in Fig. 1. As it can bee seen, the second part of an RFID system is the reader, located in between the tags and the server, who is responsible for handing over the exchanged messages between the server and a particular tag. Due to the essence of RFID tags and nature of wireless connection between the tags and the readers, all transmitted messages over these channels can be eavesdropped by an attacker. But security of communication channels between the readers and the server depends on the type of reader. In general, there are two kinds of RFID readers: *fixed* readers and *mobile* ones. In an RFID system with fixed readers, communication channels between the readers and the server are usually secure. However, in the second case where the reader is mobile and has wireless communications with the server, all communication channels between the three parties are insecure and an adversary can access all the broadcasted messages over these channels. In several cases, it has been shown that an inadequate design and an inappropriate implementation of an RFID system may enable an attacker to cause a lot of serious security and privacy concerns for RFID end-users [8]-[9].

1.1 RFID in IoT

Recently, tremendous potential of RFID systems has turned them into a popular option for different environments of IoT [4,10,11,12,13,14]. The IoT paradigm provides a huge network of connected devices in wide range of environments and infrastructures such as health-care, inventory management, smart homes, vehicular networks, transportation, monitoring and management of different infrastructures [15]. In the IoT, all objects and devices of our surrounding are connected to each other and provide a smart network without human interaction. These human-independent connections, significantly decrease errors and risks caused by exhaustion or negligence and provide benefits such as remote accessibility or addressing capability.

Recently, a lot of practical and empirical RFID-based scenarios have been proposed to be employed in different infrastructures of IoT [10,12,16,17,18]. A supplementary scenario for the GPS systems based on RFID systems can be found in [18], which could provide more accurate and low-cost location service. A recent survey on the practical RFID-based scenarios in the health-care services and environments can be found in [10]. In the survey, the authors have considered several realistic scenarios such as remote monitoring of a person's vital signs and place of residence conditions, which use RFID tags and readers as a smart and low-cost device to collect and process data. Fig. 2 shows a practical RFID-based night-care system which has recently been developed to be used in variety of care applications such as remote care of elders and children [10]. The night-care system makes it possible to have a real-time access to the collected and processed data of the under-care person on smart phones, tablets or personal computers. This capability makes it a popular system in health-care environments of the IoT. We refer to [19] for further discussion about remote patient monitoring systems. Another RFID-based practical microcosm for IoT, can be found in the *Department of Computer Science and Engineering* at the University of Washington [16]. They have used a large number of RFID tags conforming to the EPC Class-1 Generation-2 (EPC C1 G2) standard and 44 RFID readers in a 8000-Sq-m building. This project shows a successful implementation of RFID systems in IoT paradigm; more details about in [16].

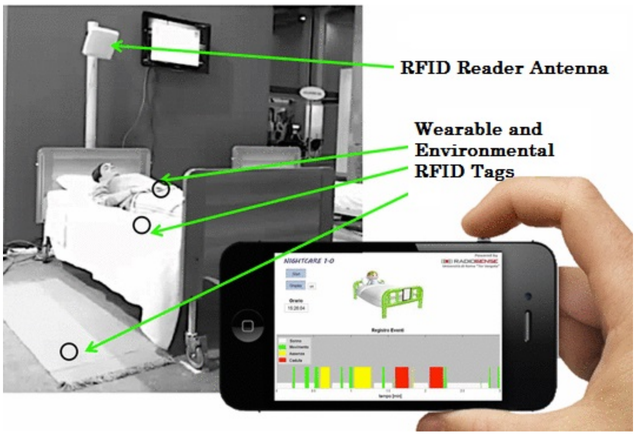


Fig. 2: The night-care system, an ambient RFID-based intelligence system for different health-care purposes in the IoT [10].

Appearance of RFID systems in a wide-range of sensitive applications has made some security and privacy concerns. Although RFID systems provide low-cost, fast, and user-friendly services, most of the users are not aware of the value of the information they broadcast when using these systems in their daily routines. In order to deal with these challenges and provide security and privacy requirements of RFID users, a lot of security schemes have been proposed, all of which try to provide a secure and anonymous authentication for end-users [20,21,22,23,24,25,26,27,28,29]. Unfortunately, design of such protocols is still an ad-hoc procedure and is based on heuristic use of different cryptographic techniques and objects. As a result, in many cases, shortly after the publication of the proposals, several attacks are found on the protocols (e.g., see [7,30,31,32,33,34,35]). These attacks, that violate the security of the protocols and make them vulnerable against different sorts of traceability attacks, are mainly due to the fact that the proposed protocols have not been designed properly and their structures bear serious problems.

1.2 Related works

A large number of endeavors have been undertaken in order to present efficient and secure authentication protocols for RFID systems. In the rest of subsection, we introduce some efforts of this domain.

In 2001, the HB and HB+ were introduced by Hopper and Blum [36], which are two authentication protocols for cheap RFID systems. Both of these protocols have been designed with a provable security paradigm in mind based on the hardness of LPN (*Learning Parity with Noise*) problem which is proved to be NP-hard. Nevertheless, a few years after their proposal, it was shown that they had several serious problems and then some improved versions of the HB-family were proposed, e.g., see [37,38,39].

Another family of RFID authentication protocols includes UMAP, LMAP, M2AP and EMAP [40,41,42], designed in 2006, which are *Ultra-lightweight*, *Light-weight*, *Minimalist*, and *Efficient* mutual authentication protocols, respectively. All these schemes have two features in common i) they have been designed to be employed in RFID tags which have lower computational capabilities and, ii) they use lightweight operations, e.g., XOR, AND, OR and etc. However, a year after their proposal, it was shown that all these protocols are vulnerable to several attacks including Denial of Service (DoS) [43,44]. In the same year, the SASI protocol [45] was proposed and was claimed to provide *Strong Authentication and Strong Integrity.* Yet, just in few months, it was broken by a series of cryptanalytic attacks, see [46,47,48].

In 2007, simultaneously with the introduction of the SASI protocol, an improved version of *Duc et al.*'s protocol [49] and the *KN* protocol [50] was proposed by *Chien* and *Chen* in [51]. In 2010, it was shown by *Yeh et al.* that the new protocols were suffering from traceability and DoS attack. This was accompanied by some fixes, which was later broken by *Yoon* [52] in 2012. *Yoon* also suggested a modified protocol whose security had been analyzed based on the hardness of finding preimages for a secure hash function $H(\cdot)$; which means for a given x computing $y = H(x)$ is easy but for a given y computing any x' such that $H(x') = y$ is hard. Nevertheless, there are several flaws in *Yoon*'s protocol that were reported by *Baghery et al.* [7], mainly due to dependency between two consecutive responses of an specific tag. In particular, all kinds of traceability attacks were mounted against the *Yoon*'s protocol and a modified version of it was presented. This version was claimed to prevent the proposed attacks. To the best of our knowledge, *Baghery et al.*'s protocol is the final and still unbroken version of this family of RFID authentication protocols, designed under the EPC C1 G2 standard [7].

Some recent articles [20,22,53,54] have focused on proposing RFID authentication protocols based on hash functions to protect the exchanged messages between the three RFID parties and to provide strong anonymity for the tag. *Ha et al.*'s protocol [53] was claimed to be anonymous and secure against various attacks. However, in 2012, *Sun* and *Zhang* discovered a weakness in the protocol, showing its incapability of providing forward privacy. *Sun* and *Zhang* also presented a novel version of the protocol and based on their analysis of the modified protocol on *Random Oracle (RO)* model, they have claimed that the improved protocol is a secure and wide-destructive private scheme in the RO model. However, the result of [31] shows that the modified version fails to achieve this security requirement too. In 2013, *Jung et al.* [54] investigated three hash-based RFID schemes and presented a novel and efficient one. Recently, in [7], it has been shown that *Jung et al.*'s protocol suffers from DoS, traceability and forward traceability attack.

Recently in [55], *Chen et al.* tried to provide a secure and untraceable authentication for RFID users using symmetric encryption and decryption functions and proposed a novel scheme. But, shortly after, *Safkhani et al.* [56] showed that they were not successful in protecting RFID users as their scheme was vulnerable to several attacks including DoS, impersonation and traceability. In order to prevent the mentioned attacks, *Safkhani et al.* applied some changes in the structure of the original protocol and claimed that the modified version eliminates all weaknesses of the original one. No surprise that the findings of [31] confirms that the modified scheme is not secure due to various traceability attacks.

Note that the main goal of all the introduced and reviewed literature was presenting a secure and confidential authentication protocol for the case that each time just one of the tags in the system has communication with the reader and the back-end server. But from an extended point of view, in order to provide private and fast authentication in large-scale RFID systems, there have been two main proposals by researchers including tree-based approaches and group-based authentications. More discussion regarding the security and privacy challenges in the large-scale RFID systems could be found in [57,58,59,60] and some research that have focused on security of low-resource devices from different perspective can be found in [61,62].

1.3 Our Contribution

The main contribution of this paper is breaking anonymity of three novel authentication protocols [25,26,28], all of which were claimed to provide strong security and anonymity for RFID end-users. Whenever applicable, we mount our attack in the well-known *Ouafi-Phan* [63] privacy model. To the best of our knowledge, all the presented security and privacy analysis of this paper are novel and this is the first time that these analyses are presented against the studied protocols. In the following, we highlight our results against the three studied protocols.

***Results on* GH** In 2015, *Gope* and *Hwang* proposed GH protocol [25], which is a realistic lightweight authentication protocol for RFID systems and was claimed to provide strong anonymity for end-users.

In the GH protocol, *Gope* and *Hwang* have tried to protect the exchanged messages using hash functions. Based on this issue, they have claimed that the tag's unique identifier ID is hidden during protocol execution. In addition, since all tag parameters are updated after each session, they are unpredictable in future runs of protocol. It is then concluded that the GH protocol is secure against forward tractability attack, meaning that it prevents tracing a specific tag in upcoming runs.

However, as a first contribution of the paper, we show that these claims and their analysis are no longer valid and GH protocol does not provide an anonymous communication for RFID end-users. More precisely, we present two different formal traceability attacks against GH protocol and prove that an attacker can trace the location of a particular tag.

***Results on* NZCL** In some applications such as inventory management, RFID mobile (handheld) readers are much more popular in comparison with fixed ones. In this case, they can be used to find a special object in a large space. Recently in [26], *Niu et al.* have proposed a novel ultra-lightweight security scheme for RFID systems with handheld readers, to which we refer as NZCL. Due to low capabilities of RFID passive tags, ultra-lightweight protocols are more practical and reasonable in most cases. This issue has motivated the designers to make their protocol more efficient and lightweight by using bitwise XOR and some lightweight PRNG functions. They have claimed that beside lightweight property, their protocol is able to protect RFID end-users against different security and privacy attacks. Some results of this study show that NZCL is unable to provide all security and privacy requirements of RFID users. Detailed analysis shows that NZCL not only suffers from secret parameters reveal, tag impersonation and Denial of Service (DoS) attacks, but also fails to provide anonymity of end-users and it is vulnerable to forward traceability attack.

***Results on* SLAP** During the last decade, providing an ultra-lightweight and secure RFID authentication protocol has been one of the challenging and popular open problems for researchers of this area. SLAP (Succinct and Lightweight Authentication Protocol) is another recently proposed ultra-lightweight RFID authentication protocol [28]. Similar to other ultra-lightweight protocols, SLAP tries to deal with security and privacy concerns of RFID users, but only using lightweight operations such as XOR, rotation and the *Conversion* which is a new lightweight operator, and more details about its definition can be found in section 3.2 of [28]. *Luo et al.* have compared SLAP with some ultra-lightweight authentication protocols and claimed that their protocol not only is more efficient in terms of total exchanged messages and tag's generated messages, but can also provide strong security and anonymity. Based on the new conversion operator, *Luo et al.* claimed that SLAP has several prominent properties such as irreversibility, sensibility and resistance against all security and privacy attacks, making it more efficient and eminent in comparison with its competitors. These claims, motivated us to analyze the structure of SLAP from different point of views and evaluate it against various security and privacy concerns. We discovered that there still are some serious drawbacks in the structure of SLAP protocol, making it undesirable in applications which need strong privacy and anonymity. We show that an attacker can use the mentioned weaknesses and perform traceability and forward traceability attacks.

1.4 Outline

The structure of this paper is as follows. Section 2 presents preliminaries and some background of RFID systems. Also, in Section 2, we will summarize *Ouafi-Phan* RFID formal privacy model which is used in our privacy analysis. GH protocol and its analysis are reported in Section 3. In Section 4, NZCL and its weaknesses are discussed. The SLAP protocol and our attacks against this protocol are considered in Section 5. We present a summary of our result in Section 6. Finally, we conclude the paper in Section 7.

2 Preliminaries

This section delineates some background of RFID systems which are used in the rest of paper and also we tabularize our notations for ease of reference.

2.1 Threats of RFID Systems

This subsection presents a short explanation of some security and privacy threats against RFID authentication protocols which are used in the next sections.

Backward Traceability In an RFID authentication protocol, it is necessary that if an attacker corrupted the secret keys of a specific tag and eavesdropped the exchanged messages between the tag and a reader, he/she should not be able to trace the location of a the target tag in the prior challenges; this property is defined as backward untraceability. This notion can be obtained by properly updating the target tag's keys [7].

Forward Traceability If an RFID authentication is not secure against forward traceability attack, it means that if an attacker corrupts secret keys of a specific tag, he/she can track the location of an end-user in his/her future executions [7].

Traceability Similar to other cryptographic protocols, in an RFID authentication protocol, it is essential that an attacker would not be able to trace a specific tag in next executions if he/she has had access to the already exchanged messages between the tag and a valid reader. This notion is known as untraceability. To this end, tags' responses in each new challange should be independent from its previous responses [7]. Fig. 3 separates the three possible privacy attacks based on the execution round of the protocol. In Fig. 3, the current round is shown with index i.

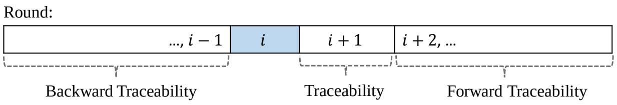


Fig. 3: Different privacy attacks in *Ouafi-Phan* formal privacy model.

Secret Parameters Reveal Protecting secret values of parties is one of the primal targets of each cryptographic protocol. In RFID cases, if an RFID authentication protocol is not designed properly, an attacker can obtain the tag secret parameters with an efficient algorithm which leads to compromise the privacy and security of parties.

Denial of Service (DoS) In the DoS attack, an end-user is deprived of the daily service which he/she expects to receive. Similarly, in RFID systems, an attacker tries to create desynchronization between a specific tag and the back-end server which makes the target tag out of service. This attack can be done in different approaches; see [64] for more discussion and approaches of applying DoS attacks.

Impersonation Attacks In the tag or reader impersonation attacks, an attacker successfully pretends the identity of the legitimate tag or reader. More precisely, an attacker tries to impersonate a legitimate tag or reader and uses the forged party capabilities to penetrate to the back-end server or to get data from it; see [64] for two sample impersonation attacks.

2.2 An RFID Formal Privacy Model

Basically, privacy of RFID security schemes can be analyzed in *ad-hoc* and *formal* approaches [65]. In the earlier approach, presented analyses are based on the notations which are defined by the attacker and do not have a special framework; that means, the adversary uses informal methods in his/her operations and analyses which are not as valid as formal methods [66]. On the other hand, in the formal approaches, the attacker's abilities are defined in specific queries which can be used in passive and active attacks. In many cases, it has been shown that in order to get a comprehensive analysis of security schemes and discover all possible weaknesses, it is necessary to use a formal privacy model [67]. Different simulation-based and game-based RFID privacy models have been proposed [63, 65, 66, 68, 69, 70, 71]. In 2008, *Ouafi* and *Phan* [63] presented a game-based formal privacy model for RFID systems which is a modified version of *Juels* and *Weis*'s privacy model [69]. Due to widespread and useful queries, this model is one of the most popular models for privacy analysis of RFID systems [32, 72, 73, 74]. We will present our privacy analysis based on this model, summarized as follows.

In this model, the attacker $\mathcal{A}$ can eavesdrop on all communication channels between parties and it can also perform active and passive attacks against them. Furthermore, the attacker $\mathcal{A}$ is allowed to run the following queries:

- Execute(R, T, i): This query models passive attacks, meaning that the attacker can eavesdrop on all transmitted messages between the tag T and the reader R in the ith session.
- Send(R, T, m, i): This query models an active attack in RFID systems. In this query, the attacker $\mathcal{A}$ has permission to impersonate the reader R in the ith session, and forwards the message m to the tag T. Note that with this query the attacker $\mathcal{A}$ has permission to alert or block the exchanged message m between the tag and the reader.
- Corrupt(T, K'): In this query, the attacker $\mathcal{A}$ is allowed to learn the stored secret key K' of the tag T. In fact, the attacker $\mathcal{A}$ has physical access to the

tag's database. Moreover, the attacker $\mathcal{A}$ can set the secret key to any arbitrary key K.

- TEST(T_0, T_1, i): This query provides the definition of untraceable privacy based on indistinguishability. If the party has accepted (see "partnership" defined below) and is being asked for a TEST query, next according to a randomly chosen bit $b \in \{0,1\}$, the attacker $\mathcal{A}$ is given the tag T_b. Now, the attacker succeeds if he/she can guess the bit b correctly. Note that in order for the notation to have a sensible meaning, a TEST session must be "fresh" as defined bellow.

Partnership A reader instance R_j and a tag instance T_i are partners if, and only if, both output ACCEPT(T_i) and ACCEPT(R_j) respectively, signifying the completion of the protocol session.

Freshness A party instance is fresh at the end of execution if, and only if: i) it outputs ACCEPT with or without a partner instance, ii) both the instance and its partner instance (if such a partner exists) have not received a CORRUPT query.

Untraceability privacy ($UPriv$) Untraceability privacy is defined by the game G that is played between an attacker $\mathcal{A}$, a reader instance and a set of tag instances. In other words, an attacker $\mathcal{A}$ plays game G using collected instances of the reader and the tag. The game G can be played using the mentioned queries in three phases as follows.

1. **Learning phase:** $\mathcal{A}$ is allowed to send each of the EXECUTE, SEND and CORRUPT queries, and interact with the reader R and randomly chosen T_0, T_1.
2. **Challenge phase:** During G, at some points the attacker $\mathcal{A}$ will select a fresh session and send a TEST query corresponding to the test session. The attacker $\mathcal{A}$ then gets a tag T_b from the set $\{T_0, T_1\}$, based on randomly chosen bit $b \in \{0,1\}$. The attacker $\mathcal{A}$ continues making any EXECUTE, SEND and CORRUPT queries as long as the *freshness* property is not violated.
3. **Guess phase:** Eventually, the attacker $\mathcal{A}$ finishes the game G and outputs a bit $b' \in \{0,1\}$ as a guess of b.

The success of attacker $\mathcal{A}$ in game G and consequently breaking the notion of $UPriv$ is quantified via $\mathcal{A}$'s advantage, denoted by $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UPriv}}(\kappa)$, in recognizing whether the attacker $\mathcal{A}$ received T_0 or T_1, which is defined as follows:

$$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UPriv}}(\kappa) = |\Pr[b' = b] - \frac{1}{2}| \; .$$

Here, κ is the security parameter and it holds that $0 \leq \mathrm{Adv}_{\mathcal{A}}^{\mathrm{UPriv}}(\kappa) \leq \frac{1}{2}$. The protocol is said to be untraceable if and only if $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UPriv}}(\kappa)$ is negligible in security parameter.

2.3 Notations

When describing the studied protocols, we stick to the notations used in the original papers. For ease of reference, notations are summarized in Table 1. To refer to the value of a variable, e.g. K_T, in the ith round, we use i as a superscript, i.e., K_T^i.

3 Analysis of GH Protocol

In order to provide security and privacy requirements of RFID end-users, impressive endeavors have been undertaken. All these efforts have had the same goals, but they have tried to reach the goals using different cryptographic techniques such as symmetric key encryption, one-way encryption and etc [20,25,26]. In this contest, one of the efficient mutual RFID authentication protocols is GH protocol which has been recently proposed by *Gope* and *Hwang* [25]. GH protocol uses one-way hash functions to protect the exchanged messages between three parties of an RFID system including the tag, the reader and the server and it has been claimed that the protocol provides strong privacy and anonymity for RFID end-users. However, our analysis and investigations on the structure of this protocol show that GH protocol does not provide end-user's privacy completely and it has some weaknesses yet, making it vulnerable to two privacy attacks. This section aims to introduce GH protocol and presents our analysis on this protocol.

3.1 GH Protocol

This section provides a review of GH protocol proposed in [25] by Gope and Hwang. The security of GH proto-

Table 1: Notations and pre-defined functions

Notation	Description
GH	The *Gope* and *Hwang*'s protocol [25]
NZCL	The *Niu et al.*'s protocol [26]
SLAP	The *Luo et al.*'s protocol [28]
$T/R/\mathcal{S}$	RFID tag/ RFID reader/ Back-end server
ID_T, IDS	Identity of the tag
AID_T	One-time-alias identity of the tag
SID	Set of shadow identity of tags
sid_j	Shadow identity of tag T_j
R_i	Identity of the ith reader
N_t, r_T	Random number generated by the tag
N_r, r_R	Random number generated by the reader
K_{ts}, K_{T_i}	Shared secret key between T and $\mathcal{S}$
K_{em}	Set of shared emergency keys between tags and server
k_{em_j}	Shared emergency key between T_j and $\mathcal{S}$
K_{rs}, K_{R_i}	Secret Key shared between the R and $\mathcal{S}$
Tr_{seq}	Track sequence number (maintain both T and $\mathcal{S}$)
$h(\cdot)$	One-way hash function
$f_K(X)$	A particular random number generator (more details in [26])
Con(X, Y)	The conversion of X and Y (it is a transformation that used in the SLAP protocol, maps two n-bit strings into an n-bit strings [28])
Rot(X, Y)	Cyclic left rotation of X based on Y's Hamming weight ([28])
$\oplus$	Exclusive-OR operation
$\|$	Concatenation operation
PRNG	Pseudo Random Number Generators
Test	Test query in *Ouafi-Phan* privacy model
Execute	Execute query in *Ouafi-Phan* privacy model
Corrupt	Corrupt query in *Ouafi-Phan* privacy model
Send	Send query in *Ouafi-Phan* privacy model
Accept	Successful verification by a tag/reader

col is based on the hardness of finding preimages for a secure hash function $H(\cdot)$. Fig 4 is taken from the original paper and illustrates the authentication procedure of GH protocol step by step. The notations used in the protocol can be found in Table 1.

Similar to other RFID authentication protocols, GH protocol has an initial or register phase and an authentication phase. In the following, we express these two phases in more details.

Initialization phase In this phase, all entities of the protocol are set to their initial values and all shared values between the server and tags take their corresponding values. In order to initialize a tag T_j, it submits its identity ID_{T_j} to the server $\mathcal{S}$, with identity ID_s, in a safe method and the server generates a random number n_s and computes $K_{ts} = h(ID_{T_j}\|n_s) \oplus ID_s$. Next, the server computes a set of shadow-IDs $SID = \{sid_1, sid_2, \cdots\}$ and a set of emergency keys $K_{em} = \{k_{em_1}, k_{em_2}, ...\}$ where $sid_j = h(ID_{T_j}\|r_j\|K_{ts})$, $k_{em_j} = h(ID_{T_j}\|sid_j\|r'_j)$ and r_j, r'_j are two random numbers. In addition, the server generates a 32-bit track sequence Tr_{seq} and sends a copy of it along with $\{K_{ts}, (SID, K_{em}), Tr_{seq}, h(\cdot)\}$ to the ID_{T_j} tag.

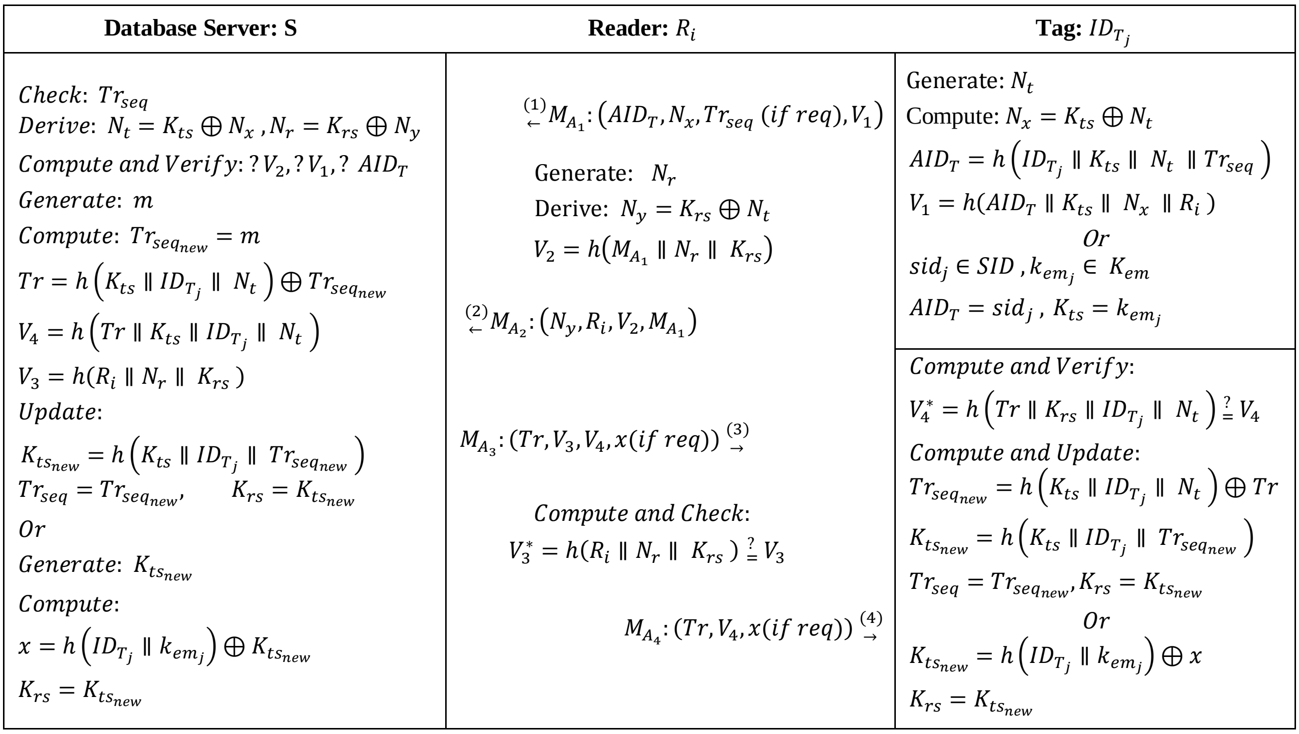
Authentication phase GH protocol is claimed to have been designed to provide a secure mutual authentication between three parties of an RFID system. To perform a complete mutual authentication, GH protocol needs five steps with following processes.

Step 1 (Tag → Reader): The tag ID_{T_j} generates a random number N_t to derive $AID_T = h(ID_{T_j}\|K_{ts}\|N_t\|Tr_{seq})$, $N_x = K_{ts} \oplus N_t$, and $V_1 = h(AID_T\|K_{ts}\|N_x\|R_i)$. Then, the tag forms a messages $M_{A_1} = \{AID_T, N_x, Tr_{seg}(\text{if req.}), V_1\}$ and sends it as a request to the reader.

Step 2 (Reader → Server): Upon receiving the tag response M_{A_1} which includes $\{AID_T, N_x, Tr_{seg}(\text{if req.}), V_1\}$, the reader generates a random number N_r, computes $N_y = K_{rs} \oplus N_r$, $V_2 = h(M_{A_1}\|N_r\|K_{rs})$ and sends them as a request message M_{A_2} to the server.

Step 3 (Server → Reader): Upon receiving the message M_{A_2} from the reader, the server checks validity of Tr_{seq} and $V_1 = h(AID_T\|K_{ts}\|N_x\|R_i)$. If they were valid, the server computes $N_t = K_{ts} \oplus N_x$ and verifies AID_T. In case they both were not valid, the server stops the protocol. Now, after successful verification of AID_T, the server generates m as a random number and sets $Tr_{seq_{new}} = m$. Next, the server calculates $Tr = h(K_{ts}\|ID_{t_j}\|N_t) \oplus Tr_{seq_{new}}$, $V_4 = h(Tr\|K_{ts}\|ID_{T_j}\|N_t)$, $V_3 = h(R_i\|N_r\|K_{rs})$ and arranges them in a message M_{A_3} and sends it to the reader. Finally, it updates its values with $K_{ts_{new}}$ and $Tr_{seq_{new}}$, where $K_{ts_{new}} = h(K_{ts}\|ID_{T_j}\| Tr_{seq_{new}})$.

Note that if the server cannot find Tr_{seq} in M_{A_1}, then it will verify the AID_T using sid_j. If it was successful, the server generates a new shared key $K_{ts_{new}}$ and computes



Database Server: S	**Reader:** R_i	**Tag:** ID_{T_j}
$Check\colon Tr_{seq}$ $Derive\colon N_t = K_{ts} \oplus N_x, N_r = K_{rs} \oplus N_y$ $Compute\ and\ Verify\colon ?V_2, ?V_1, ?\ AID_T$ $Generate\colon m$ $Compute\colon Tr_{seq_{new}} = m$ $Tr = h\left(K_{ts} \Vert ID_{T_j} \Vert N_t\right) \oplus Tr_{seq_{new}}$ $V_4 = h\left(Tr \Vert K_{ts} \Vert ID_{T_j} \Vert N_t\right)$ $V_3 = h(R_i \Vert N_r \Vert K_{rs})$ $Update\colon$ $K_{ts_{new}} = h\left(K_{ts} \Vert ID_{T_j} \Vert Tr_{seq_{new}}\right)$ $Tr_{seq} = Tr_{seq_{new}}, \quad K_{rs} = K_{ts_{new}}$ Or $Generate\colon K_{ts_{new}}$ $Compute\colon$ $x = h\left(ID_{T_j} \Vert k_{em_j}\right) \oplus K_{ts_{new}}$ $K_{rs} = K_{ts_{new}}$	$\overset{(1)}{\leftarrow} M_{A_1}\colon (AID_T, N_x, Tr_{seq}\ (if\ req), V_1)$ Generate: N_r Derive: $N_y = K_{rs} \oplus N_t$ $V_2 = h(M_{A_1} \Vert N_r \Vert K_{rs})$ $\overset{(2)}{\leftarrow} M_{A_2}\colon (N_y, R_i, V_2, M_{A_1})$ $M_{A_3}\colon (Tr, V_3, V_4, x(if\ req)) \overset{(3)}{\rightarrow}$ $Compute\ and\ Check\colon$ $V_3^* = h(R_i \Vert N_r \Vert K_{rs}) \overset{?}{=} V_3$ $M_{A_4}\colon (Tr, V_4, x(if\ req)) \overset{(4)}{\rightarrow}$	Generate: N_t Compute: $N_x = K_{ts} \oplus N_t$ $AID_T = h\left(ID_{T_j} \Vert K_{ts} \Vert N_t \Vert Tr_{seq}\right)$ $V_1 = h(AID_T \Vert K_{ts} \Vert N_x \Vert R_i)$ Or $sid_j \in SID, k_{em_j} \in K_{em}$ $AID_T = sid_j, K_{ts} = k_{em_j}$ $Compute\ and\ Verify\colon$ $V_4^* = h\left(Tr \Vert K_{rs} \Vert ID_{T_j} \Vert N_t\right) \overset{?}{=} V_4$ $Compute\ and\ Update\colon$ $Tr_{seq_{new}} = h\left(K_{ts} \Vert ID_{T_j} \Vert N_t\right) \oplus Tr$ $K_{ts_{new}} = h\left(K_{ts} \Vert ID_{T_j} \Vert Tr_{seq_{new}}\right)$ $Tr_{seq} = Tr_{seq_{new}}, K_{rs} = K_{ts_{new}}$ Or $K_{ts_{new}} = h\left(ID_{T_j} \Vert k_{em_j}\right) \oplus x$ $K_{rs} = K_{ts_{new}}$

Fig. 4: GH protocol [25]

the message x as $x = K_{ts_{new}} \oplus h(ID_{T_j} \| k_{em_j})$ by the emergency key k_{em_j}. In this case the server also includes x in the response message M_{A_3} and sends it to the reader.

Step 4 (Server → Tag): Upon receiving M_{A_3} from the server, the reader calculates $h(R_i \| N_r \| K_{rs})$ and verifies whether $V_3 \overset{?}{=} h(R_i \| N_r \| K_{rs})$. If the answer was YES, the reader constructs $M_{A_4} : \{Tr, V_4, x(\text{if req.})\}$ and sends it to the tag.

Step 5 (Tag): Finally the tag computes $h(Tr \| K_{ts} \| ID_{T_j} \| N_t)$ and verifies whether $V_4 \overset{?}{=} h(Tr \| K_{ts} \| ID_{T_j} \| N_t)$. If so, it derives $Tr_{seq_{new}} = h(K_{ts} \| ID_{T_j} \| N_t) \oplus Tr$, $K_{ts_{new}} = h(K_{ts} \| ID_{T_j} \| Tr_{seq_{new}})$ and saves $Tr_{seq} = Tr_{seq_{new}}$, $K_{ts} = K_{ts_{new}}$ for further authentications. Otherwise, the tag initiates a new request with an unused pair of secret identity and emergency key.

3.2 Traceability Attack

Protecting the privacy of end-users is one of the main goals of each authentication protocol. In this subsection, we aim to show that GH protocol does not protect RFID users against traceability attack and a malicious attacker can trace the location of a specific tag as follows.

Learning phase In the ith round, the attacker $\mathcal{A}$ sends an $\textsc{Execute}(R, T_0, i)$ query and obtains Tr_{seq,T_0} from the tag T_0.

Challenge phase The attacker $\mathcal{A}$ selects two fresh tags T_0 and T_1 for test, and sends a $\textsc{Test}(T_0, T_1, i+1)$ query. According to the randomly chosen bit $b \in \{0,\ 1\}$, the attacker is given a tag T_b. After that, the attacker $\mathcal{A}$ sends an $\textsc{Execute}(R, T_0, i+1)$ query, and he/she obtains Tr_{seq,T_b}.

Guess phase The attacker $\mathcal{A}$ stops the game G and outputs a bit $b' \in \{0,\ 1\}$ as a guess for b. In order to determine $b' \in \{0,\ 1\}$, the attacker uses the following rule:

$$b' = \begin{cases} 0 & \text{if } Tr_{seq,T_0} = Tr_{seq,T_b} \\ 1 & \text{otherwise} \end{cases} .$$

Claim The GH protocol fails to provide untraceability and, in particular, for the constructed attacker $\mathcal{A}$, we have $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UPriv}}(\kappa) = \frac{1}{2}$.

Proof According to the updating procedure of GH protocol, we have $Tr_{seq_{new}} \leftarrow h(K_{ts}\|ID_{T_j}\|N_t) \oplus Tr$ and the tag T_0 does not update its secret values in the *Learning* phase and uses the same secret value Tr_{seq} in both *Learning* and *Challenge* phases. As a result, if Tr_{seq,T_0}=Tr_{seq,T_b} then the selected tag T_b is exactly the target tag T_0. Therefore, $b' = b$ with probability one and hence $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UPriv}}(\kappa) = |1-\frac{1}{2}| = \frac{1}{2}$, meaning that the protocol is traceable.

3.3 Forward Traceability Attack

This section shows that there is another privacy concern in the GH protocol which is vulnerability against forward traceability attack. This weakness is caused due to a flaw in updating the secret key K_{ts} and fixing secret value ID_{T_j}. By considering this fact, an attacker can obtain K_{ts}^{i+1} to track. More details follow.

Learning phase In the ith round, the attacker $\mathcal{A}$ sends a $\textsc{Corrupt}(T_0, K')$ query and obtains K_{ts,T_0}^{i} and $ID_{T_0}^{i}$ from the tag T_0.

Challenge phase The attacker $\mathcal{A}$ selects two fresh tags T_0 and T_1 for test, and sends a $\textsc{Test}(T_0, T_1, i)$ query. According to the randomly chosen bit $b \in \{0,\ 1\}$, the attacker is given the tag T_b and then, in $(i+1)$th round, the attacker $\mathcal{A}$ sends an $\textsc{Execute}(R, T_b, i+1)$ query, and obtains Tr_{seq,T_b}^{i+1}, $AID_{T_b}^{i+1}$ and N_{x,T_b}^{i+1}. Then, he/she computes $K_{ts}^{\mathrm{att}} = h(K_{ts,T_0}^{i}\|ID_{T_0}^{i}\|Tr_{seq,T_b}^{i+1})$, $N_t^{\mathrm{att}} = N_{x,T_b}^{i+1} \oplus K_{ts,T_0}^{i}$ and $AID_{T_0}^{\mathrm{att}} = h(ID_{T_0}^{i}\|K_{ts}^{\mathrm{att}}\|N_t^{\mathrm{att}}\|Tr_{seq,T_b}^{i+1})$.

Guess phase The attacker $\mathcal{A}$ stops the game G and outputs a bit $b' \in \{0,\ 1\}$ as a guess for b. In order to determine $b' \in \{0,\ 1\}$, the attacker uses the following rule:

$$b' = \begin{cases} 0 & \text{if } AID_{T_b}^{i+1} = AID_{T_0}^{\mathrm{att}} \\ 1 & \text{otherwise} \end{cases} .$$

Claim The GH protocol fails to provide forward untraceability and, in particular, for the constructed attacker $\mathcal{A}$, we have $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UPriv}}(\kappa) = \frac{1}{2}$.

Proof According to the updating procedure and tag response of GH protocol we have:

$$\begin{aligned} AID_{T_0}^{\mathrm{att}} &= h(ID_{T_0}^{i}\|K_{ts}^{\mathrm{att}}\|N_t^{\mathrm{att}}\|Tr_{seq,T_b}^{i+1}) \\ &= h(ID_{T_0}^{i}\|h(K_{ts,T_0}^{i}\|ID_{T_0}^{i}\|Tr_{seq,T_b}^{i+1})\|N_t^{\mathrm{att}}\| \\ &\quad Tr_{seq,T_b}^{i+1}) \\ &= h(ID_{T_0}^{i}\|h(K_{ts,T_0}^{i}\|ID_{T_0}^{i}\|Tr_{seq,T_b}^{i+1})\|N_{x,T_b}^{i+1} \\ &\quad \oplus K_{ts,T_0}^{i}\|Tr_{seq,T_b}^{i+1}) \ , \end{aligned}$$

where the simplifications are due to the relations $K_{ts}^{\mathrm{att}} = h(K_{ts,T_0}^{i}\|ID_{T_0}^{i}\|Tr_{seq,T_b}^{i+1})$ and $N_t^{\mathrm{att}} = N_{x,T_b}^{i+1} \oplus K_{ts,T_0}^{i}$. Now, if $T_b = T_0$ then due to fixing ID_{T_j} and using the relation $N_t^{i+1} = N_{x,T_b}^{i+1} \oplus K_{ts,T_b}^{i}$, we get:

$$\begin{aligned} AID_{T_0}^{\mathrm{att}} &= h(ID_{T_b}^{i}\|h(K_{ts,T_b}^{i}\|ID_{T_b}^{i}\|Tr_{seq,T_b}^{i+1})\|N_{x,T_b}^{i+1} \\ &\quad \oplus K_{ts,T_b}^{i}\|Tr_{seq,T_b}^{i+1}) \\ &= h(ID_{T_b}^{i}\|h(K_{ts,T_b}^{i}\|ID_{T_b}^{i}\|Tr_{seq,T_b}^{i+1})\|N_{t,T_b}^{i+1}\|Tr_{seq,T_b}^{i+1}) \\ &= AID_{T_b}^{i+1} \ . \end{aligned}$$

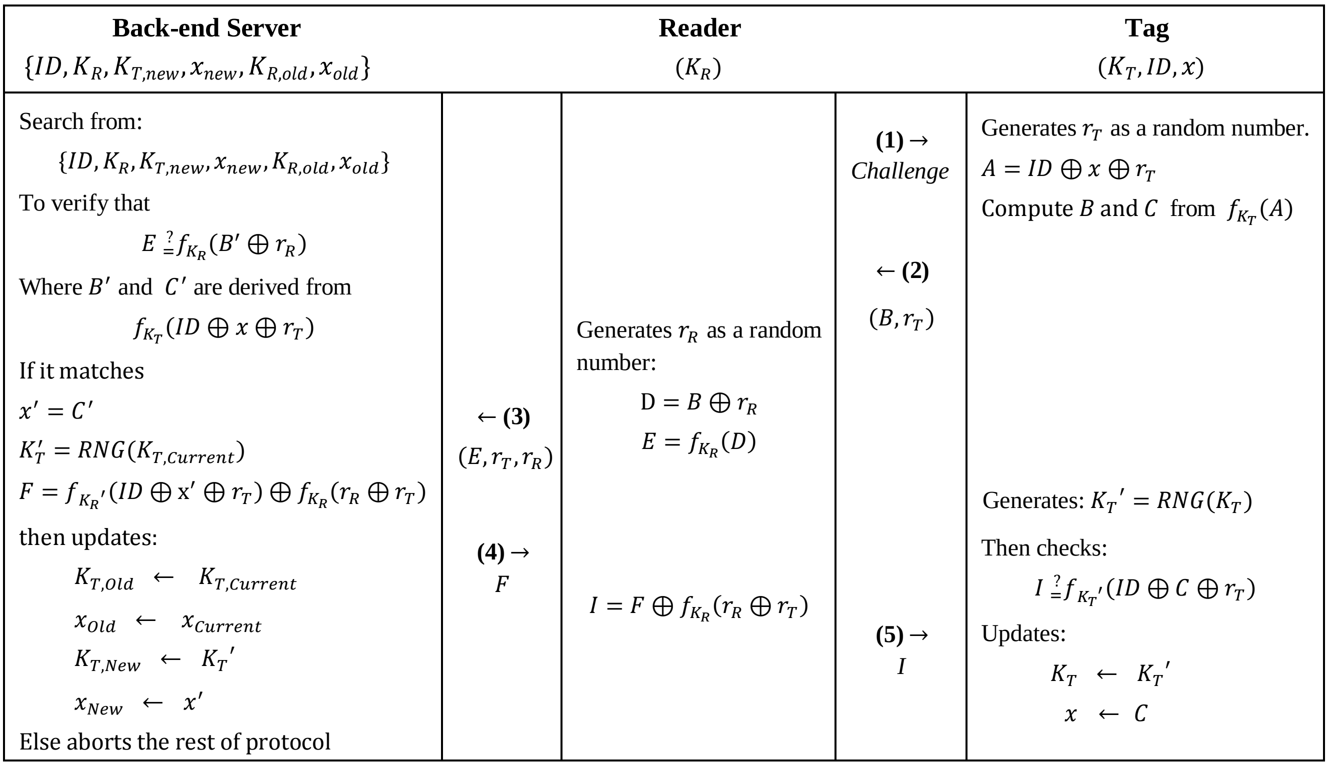
Therefore, $b' = b$ with probability one and hence $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UPriv}}$ $(\kappa) = |1-\frac{1}{2}| = \frac{1}{2}$ which means that the target tag is forward traceable.

4 Analysis of NZCL

NZCL is yet another authentication protocol for RFID systems with mobile reader, proposed by *Niu et al.* [26]. In the RFID systems with mobile reader, the communication channel between the reader and the server is insecure. In this section, in order to get more familiar with NZCL, first we introduce it and then present our security and privacy analysis against this protocol. We present our privacy analysis based on *Ouafi-Phan* formal privacy model which is summarized in Section 2.2.

4.1 Description of NZCL

The structure of NZCL [26] has been shown in Fig. 5, taken from the original paper. NZCL is a lightweight protocol as it mainly uses operations such as bitwise XOR and RNG functions and its security is based on the fact that $f_K(X)$ is a pseudo random number generator (RNG). Similar to majority of the RFID authentication protocols, NZCL has two main phases, *Initialization* and *Authentication*, summarized as follows.



Back-end Server $\{ID, K_R, K_{T,new}, x_{new}, K_{R,old}, x_{old}\}$		**Reader** (K_R)		**Tag** (K_T, ID, x)
Search from: $\{ID, K_R, K_{T,new}, x_{new}, K_{R,old}, x_{old}\}$ To verify that $E \overset{?}{=} f_{K_R}(B' \oplus r_R)$ Where B' and C' are derived from $f_{K_T}(ID \oplus x \oplus r_T)$			**(1)** → *Challenge* ← **(2)** (B, r_T)	Generates r_T as a random number. $A = ID \oplus x \oplus r_T$ Compute B and C from $f_{K_T}(A)$
If it matches $x' = C'$ $K'_T = RNG(K_{T,Current})$ $F = f_{K_R{'}}(ID \oplus x' \oplus r_T) \oplus f_{K_R}(r_R \oplus r_T)$	← **(3)** (E, r_T, r_R)	Generates r_R as a random number: $D = B \oplus r_R$ $E = f_{K_R}(D)$		
then updates: $K_{T,Old} \leftarrow K_{T,Current}$ $x_{Old} \leftarrow x_{Current}$ $K_{T,New} \leftarrow K_T{'}$ $x_{New} \leftarrow x'$ Else aborts the rest of protocol	**(4)** → F	$I = F \oplus f_{K_R}(r_R \oplus r_T)$	**(5)** → I	Generates: $K_T{'} = RNG(K_T)$ Then checks: $I \overset{?}{=} f_{K_T{'}}(ID \oplus C \oplus r_T)$ Updates: $K_T \leftarrow K_T{'}$ $x \leftarrow C$

Fig. 5: NZCL RFID authentication protocol [26]

Initialization phase In this phase, all entities of the protocol are reset to their initial values. The back-end server shares keys K_R and $\{ID, K_T, x\}$ with the reader and the tag, respectively. Here, ID is identifier of the tag, K_T and K_R are L-bit long secret keys of the tag and the reader, where typically $L = 16$, and x is a shared key between the server and tag. All the mentioned values are updated and that all get new values after each successful run of the protocol.

Authentication phase NZCL is a mutual authentication protocol and its authentication phase consists of five steps which can be expressed as follows:

Step 1 (Reader → Tag): The reader starts new session with the tag by sending a random number *Challenge* to the tag.

Step 2 (Tag → Reader): First the tag generates the random number r_T, and computes $A = ID \oplus x \oplus r_T$ and obtains the pseudo-random strings B and C from the pseudo-random function $f_{K_T}(A)$ (which probably means $B||C = f_{K_T}(A)$). Finally, the tag sends r_T and B to the reader.

Step 3 (Reader → Back-end server): Similar to the previous step, the reader generates a random number r_R and then computes $D = B \oplus r_R$ and $E = f_{K_R}(D)$. Then, the reader sends E along with r_R and r_T to the back-end server.

Step 4 (Back-end server → Reader): The back-end server uses the received messages from the reader and acts as follows:

- It retrieves B' and C' from $f_{K_T}(ID \oplus x \oplus r_T)$.
- Then, it searches the database and finds a tuple $\{ID, K_R, K_T, x\}$ which satisfies $E = f_{K_R}(B' \oplus r_R)$.
- It pre-updates $x' = C'$ and $K'_T = RNG(K_T)$ if the mentioned relation is verified successfully. Then, it computes $F = f_{K'_T}(x' \oplus r_T) \oplus f_{K_R}(r_R \oplus r_T)$ and sends it to the reader.
- Finally, it updates the values of x' and K'_T as $x' \leftarrow C'$ and $K'_T \leftarrow RNG(K_T)$.

Step 5 (Reader → Tag): The reader uses the received message F from the server and computes $I = F \oplus f_{K_R}(r_R \oplus r_T)$ and sends it to the tag.

Step 6 (Tag): Finally, the tag authenticates the reader and the server by comparing the message I with $f_{K'_T}(C \oplus r_T)$ where $K'_T = RNG(K_T)$. At the end, it updates its values x and K_T as $x \leftarrow C$ and $K_T \leftarrow K'_T$.

4.2 Secret Parameters Reveal

In this subsection, we show that NZCL does not protect secret keys properly and an attacker is able to reveal the secret parameter K_R. Our attacker simply performs a brute-force search based on the eavesdropped values to recover the secret key K_R as follows:

Learning phase In this phase, the attacker acts as an eavesdropper. After one successful run, he/she saves the exchanged data between the target tag and the server including r_R, $E = f_{K_R}(D)$ and B.

Attack phase Recall that K_R is an L-bit long string, and L is typically chosen to be 16. Based on the obtained values r_R, $E = f_{K_R}(D)$ and B in the learning phase, and due to the relation $D = B \oplus r_R$, it holds that $E = f_{K_R}(B \oplus r_R)$. Since there are only 2^L possibilities for K_R, this secret value can then be determined using a simple exhaustive search. Note that in order to provide a stronger security level, one can increase the complexity of this attack to $O(2^{2L})$ which is the optimal security bound, see [75] for more discussion.

Note that after performing this attack and obtaining the secret value of the reader, the attacker can perform reader traceability attack, and reader impersonation attack with the success probability of "1". Furthermore, NZCLhas some other weaknesses, due to which in the rest of paper we present several possible attacks.

4.3 Tag Impersonation and DoS Attacks

In this section, beside the discovered weakness in the last subsection, we provide a practical impersonation attack on NZCL. We illustrate that an attacker can impersonate a legitimate tag after eavesdropping one round of the protocol and starting a new challenge with the reader. It can be shown that after this impersonation attack, the attacker can desynchronize the tag and the server simply with blocking one step of the main protocol. This impersonation attack is performed in two phases as follows:

Learning phase In round i, the attacker eavesdrops the protocol and saves exchanged messages $I^i = F^i \oplus f_{K^i_R}(r^i_T \oplus r^i_R) = f_{K'^i_T}(ID \oplus x'^i \oplus r^i_T)$ and r^i_T where $K'^i_T = K^{i+1}_T$. Then, the server authenticates the tag and updates its secret parameters, the attacker terminates the rest of the session and prevents the tag from updating its secret values.

Attack phase In this phase, the attacker plays the role of a legitimate tag and starts a new session with the reader and performs the following operations in four steps.

Step 1 (Reader → Attacker): The reader sends the message *Challenge* to the attacker.

Step 2 (Attacker → Reader): In round $(i+1)$, the attacker uses the eavesdropped messages in the *Learning phase* and assumes that sets $f_{K^{i+1}_T}(A_{\text{att}})$ and $r_{T,\text{att}}$ as follows,

$$f_{K^{i+1}_T}(A_{\text{att}}) = f_{K'^i_T}(ID \oplus x'^i \oplus r^i_T) = I^i,$$
$$r_{T,\text{att}} = r_{T,i}.$$

Then he/she computes B_{att} and C_{att} from $f_{K^{i+1}_T}(A_{\text{att}}) = I^i$ and transmits messages B_{att} and $r_{T,\text{att}}$ to the reader.

Step 3 (Reader → Back-end server): The reader computes $D_{\text{att}} = B_{\text{att}} \oplus r^{i+1}_R$, and $E^{i+1} = f_{K^{i+1}_R}(B_{\text{att}} \oplus r^{i+1}_R$. Then he/she sends messages E^{i+1}, r_{att}, and r^{i+1}_R to the back-end server.

Step 4 (Back-end server): The back-end server receives the transmitted messages from the reader and follows the usual procedure of the protocol. Since, the stored secret key $K_{T,new}$ in the server is equal to K^{i+1}_T or K'^i_T, as a result $B'^i = B_{\text{att}}$ and $C'^i = C_{\text{att}}$. Therefore, the server authenticates the attacker as a legitimate tag.

4.4 Forward Traceability Attack

This section aims to show that there is another privacy weakness in NZCL which makes it vulnerable to the forward traceability attack. According to the structure of the protocol, it can be seen that the ID is fixed in all rounds. Due to this fact, an attacker can track a target tag as follows:

Learning phase In the ith round, the attacker $\mathcal{A}$ sends a $\textsc{Corrupt}(T_0, K')$ query and obtains $(K^i_{T_0}, x^i_{T_0}, ID^i_{T_0})$ from the tag T_0.

It sends an $\textsc{Execute}(R, T_0, i)$ query and obtains r^i_T. Now, the attacker can compute $K^{i+1}_{T_0}$ at the session $i+1$ by computing $\text{PRNG}(K^i_{T_0})$. Then he/she compute $\psi^{i+1} = ID^i_{T_0} \oplus x^i_{T_0} \oplus r^i_{T_0}$, consequently $B^i_{T_0}$ and $C^i_{T_0}$ can be obtained from $f_{K^i_{T_0}}(\psi^{i+1})$. Then he/she sets $\zeta = C^i_{T_0}$.

Challenge phase The attacker $\mathcal{A}$ selects two fresh tags T_0 and T_1 for the test, and sends a $\textsc{Test}(T_0, T_1, i+1)$ query. According to the randomly chosen bit $b \in \{0,\ 1\}$, the attacker is given a tag $T_b \in \{T_0,\ T_1\}$. After that, in the $(i+1)$th round, the attacker $\mathcal{A}$ sends an $\textsc{Execute}(R, T_b, i+1)$ query by sending *Challenge* and obtains $(B^{i+1}_{T_b}, r^{i+1}_{T_b})$.

Guess phase The attacker $\mathcal{A}$ stops the game G and outputs a bit $b' \in \{0,\ 1\}$ as a guess for b. In order to determine $b' \in \{0,\ 1\}$, firstly the attacker $\mathcal{A}$ computes $\psi^{i+1} = ID^i_{T_0} \oplus \zeta \oplus r^{i+1}_{T_b}$, and calculates $B^{i+1}_{T_0}$ and $C^{i+1}_{T_0}$ from $f_{K^{i+1}_{T_0}}(\psi^{i+1})$, where $K^{i+1}_{T_0} = \text{PRNG}(K^i_{T_0})$. Then, the attacker outputs a bit $b' \in \{0,\ 1\}$ as a guess for b using the following rule:

$$b' = \begin{cases} 0 & \text{if} \quad B^{i+1}_{T_b} = B^{i+1}_{T_0} \\ 1 \text{ otherwise} \end{cases} .$$

Claim The NZCL protocol fails for provide forward untraceability and, in particular, for the constructed attacker $\mathcal{A}$, we have

$$\text{Adv}^{\text{UPriv}}_{\mathcal{A}}(\kappa) = \frac{1}{2} .$$

Proof According to the authentication phase of the protocol, it can be seen that the identifier of a specific tag is fixed all the time, which means $ID^{i+1} = ID^i = ID$. Now, since the attacker uses the keys of the legitimate tag in ith phase, the server accepts him/her as a legitimate tag using old stored keys of the target tag.

5 Analysis of the SLAP Protocol

The next protocol which we analyze in this paper, is the SLAP authentication protocol [28]. Our analysis shows that the SLAP protocol does not act fine in protecting privacy of RFID end-users. Precisely, in this section, we prove that the SLAP protocol is not secure against traceability and forward traceability attacks. To this aim, we begin the section by introducing the SLAP protocol and then complete it by presenting our formal privacy analysis.

5.1 Description of SLAP

In SLAP [28], communication channel between the reader and the back-end server is secure, but all exchanged messages over communication channels between the tags and the reader can be eavesdropped by an eavesdropper. The back-end server has shared a unique code and some updatable secret keys between each tag. Tags use their corresponding keys and the identifiers during authentication process with the server. Fig. 6, taken from the original paper, shows the structure of the SLAP protocol. As it can be seen, the SLAP protocol is a mutual authentication protocol and the tag and the server/reader authenticate each other in five steps, which will be explained shortly. We remark that in the SLAP protocol Con is a transformation which stands for conversion. The conversion of two n-bit strings A and B is an n-bit string denoted by $\text{Con}(A, B)$. The designers base the security of the SLAP protocol on the following property of the function $\text{Con}(A, B)$: it is assumed that knowing one of the two inputs and the corresponding out, it is hard to find the other input. This transformation is composed of three complicated operations itself. We refer the reader to [28] (section 3.2) for detailed description of the conversion mapping. For our purpose, the details of this transformation is not relevant.

Step 1 (Reader/Server → Tag): The reader starts a new session with the tag by sending a *Hello* message to the tag.

Step 2 (Tag → Reader/Server): The tag responds to the reader with IDS, which is a pseudonym.

Step 3. (Reader/Server → Tag): The server/reader uses the received IDS and finds the corresponding entry in the database. In case $IDS = IDS^{old}$, the reader / server will use old secret keys K^{old}_1 and K^{old}_2 to calculate the transmitted messages, otherwise it uses the new

Database \ Reader		**RFID Tag**
$\{IDS_{old}, IDS_{new}, K_1^{old}, K_1^{new}, K_2^{old}, K_2^{new}\}$		$\{IDS_{old}, IDS_{new}, K_1^{old}, K_1^{new}, K_2^{old}, K_2^{new}\}$
$A = Con\,(K_1, K_2) \oplus \text{n}$ $B = Con(Rot(K_1, n), K_1 \oplus K_2)$ $\oplus\, Rot(Con(K_2, K_2 \oplus n), K_1)$	**(1)** → $Hello$	$C = Con(Con\,(B, K_1^{new}), Con(K_1^{new}, K_2^{new} \oplus n)) \oplus ID$
After receiving $C_{L\,or\,R}$, it computes $C_{L\,or\,R}$ locally to authenticate the tag, it checks whether they are equal.	← **(2)** IDS	
	(3) → $A \,\|\|\, B_{L\,or\,R}$	
Key Updating: ${K_1}^{new} \leftarrow Con\,(K_1, n) \oplus K_2$ ${K_2}^{new} \leftarrow Con\,(K_2, B) \oplus K_2$ $IDS^{new} \leftarrow Con\,(IDS, n \oplus (B''_{L\,or\,R} \,\|\|\, C''_{L\,or\,R}))$	← **(4)** $C_{L\,or\,R}$	Key Updating: ${K_1}^{new} \leftarrow Con\,(K_1, n) \oplus K_2$ ${K_2}^{new} \leftarrow Con\,(K_2, B) \oplus K_1$ $IDS^{new} \leftarrow Con\,(IDS, n \oplus (B''_{L\,or\,R} \,\|\|\, C''_{L\,or\,R}))$

Fig. 6: The SLAP protocol [28].

secret keys K_1^{new} and K_2^{new}. Note that in case there is no matching IDS, the reader/server will discard the tag and will stop the rest of protocol. After determining the IDS, the reader/server generates the pseudo random number n and computes A and B using operators $\text{Con}(X, Y)$ and $\text{Rot}(X, Y)$ (as shown in Fig. 6), where $\text{Rot}(X, Y)$ is cyclic left rotation of X based on Y's Hamming weight. Finally, the reader/server sends A and one of B_L (left half of B) or B_R (right half of B) to the reader. Selection of B_L or B_R is based on the parity of Hamming weight of B, that is, depending on whether it is odd or even, the reader/server sends B_L or B_R, respectively.

Step 4 (Reader/Server → Tag): The tag XORs A and $\text{Con}(K_1, K_2)$ and extracts the random number n. Then, it calculates the value of B' with its corresponding secret keys K_1/K_2 and then performs the operation of the last step with B' to check whether B'_{LorR} is equal to B_{LorR} or not. If the answer was YES, the tag authenticates the reader/server as a legitimate reader/server and will perform the following operations.

- It updates the secret keys K_1^{new}, K_2^{new} and IDS^{new} as

  $$K_1^{new} \leftarrow \text{Con}(K_1, n) \oplus K_2$$
  $$K_2^{new} \leftarrow \text{Con}(K_2, n) \oplus K_1$$
  $$IDS^{new} \leftarrow \text{Con}(IDS, n \oplus (B''_{LorR} \| C''_{LorR}))$$

  where B''_{LorR} and C''_{LorR} are the other half of messages B and C which have not been sent by the reader/server and the tag during steps 3 and 4, respectively.
- It calculates the message C with the updated keys (K_1^{new}, K_2^{new}) and random number n, then sends the corresponding message C_{LorR} (based on the parity of the Hamming weight) to the reader/server. If $B'_{LorR} \neq B_{LorR}$, the tag will stop the rest of protocol immediately.

Step 5 (Reader/Server): Upon receiving message C_{LorR} from the tag, the reader / server computes C' and checks whether C'_{LorR} is equal to C_{LorR}. If they are, the reader/server authenticates the tag successfully and the reader / server updates its IDS and secret keys (K_1/K_2) in the same way as the tag; otherwise, the authentication is not successful and the reader / server does not update its corresponding keys.

5.2 Traceability Attack

As we mentioned before, in an RFID authentication protocol, the tag can provide its owner privacy if its responses in two consecutive challenges are uncorrelated. In this section, we show that the SLAP protocol has not been designed efficiently and it is vulnerable to traceability attack. In order to do this attack, an attacker performs the following operations:

Learning phase In the ith round, the attacker $\mathcal{A}$ sends an $\text{Execute}(R, T_0, i)$ query and obtains $IDS^i_{T_0}$ from the tag T_0. Finally he/she sends a $\text{Send}(R, T_0, i)$

query and stops the protocol, such that, the tag cannot update its secret values.

Challenge phase The attacker $\mathcal{A}$ selects two fresh tags T_0 and T_1 for test, and sends a $\textsc{Test}(T_0, T_1, i+1)$ query. According to the randomly chosen bit $b \in \{0,\ 1\}$, the attacker is given the tag T_b. After that, the attacker $\mathcal{A}$ sends an $\textsc{Execute}(R, T_0, i+1)$ query and obtains $IDS^{i+1}_{T_b}$.

Guess phase The attacker $\mathcal{A}$ stops the game G and outputs a bit $b' \in \{0,\ 1\}$ as a guess for b, based on the following rule:

$$b' = \begin{cases} 0 & \text{if} \quad IDS^{i+1}_{T_b} = IDS^{i}_{T_0} \\ 1 \text{ otherwise} & \end{cases} .$$

Claim The SLAP protocol fails to provide untraceability and, in particular, for the constructed attacker $\mathcal{A}$, we have

$$\mathrm{Adv}^{\mathrm{UPriv}}_{\mathcal{A}}(\kappa) = \frac{1}{2} \ .$$

Proof According to the update procedure of SLAP, $IDS^{i+1} \leftarrow \mathrm{Con}(IDS^i, n \oplus (B^{''}_{LorR} \| C^{''}_{LorR})$ and the tag T_0 does not update its secret values in the *Learning* phase and uses the same secret value IDS in both *Learning* and *Challenge* phases. As a result, if $IDS^{i+1}_{T_b}$=$IDS^{i}_{T_0}$ then the selected tag T_b is exactly the target tag T_0. This is translated as $\Pr[b' = b] = 1$, from which the claim follows.

5.3 Forward Traceability Attack

In majority of applications of RFID systems, it is essential that the implemented authentication protocol can provide forward untractability for end-users (the tag). Now, we show that the SLAP protocol does not protect end-user's location in the future runs and it suffers from froward traceability attack. In order to perform this attack in *Ouafi-Phan* privacy model, an attacker acts as follows:

Learning phase In the ith round, the attacker $\mathcal{A}$ sends a $\textsc{Corrupt}(T_0, K')$ query and obtains K^i_{1,T_0}, K^i_{2,T_0} and ID_{T_0} from the tag T_0.

Challenge phase The attacker $\mathcal{A}$ selects two fresh tags T_0 and T_1 for test, and sends a $\textsc{Test}(T_0, T_1, i)$ query. According to the randomly chosen bit $b \in \{0,\ 1\}$, the attacker is given the tag T_b. After that, in the $(i+1)$th round, the attacker $\mathcal{A}$ sends an $\textsc{Execute}(R, T_b, i+1)$ query and obtains $A^{i+1}_{T_b}$ and C^{i+1}_{LorR,T_b}. Then he/she computes the values $\alpha = \mathrm{Con}(K^i_{1,T_0}, K^i_{2,T_0})$, $n^{i+1}_{\mathrm{att}} = A^{i+1}_{T_b} \oplus \alpha$ and $B^{i+1}_{\mathrm{att}} = \mathrm{Con}(\mathrm{Rot}(K^i_{1,T_0}, n^{i+1}_{\mathrm{att}}), K^i_{1,T_0} \oplus K^i_{2,T_0}) \oplus \mathrm{Rot}(\mathrm{Con}(K^i_{2,T_0}, K^i_{2,T_0} \oplus n^{i+1}_{\mathrm{att}}), K^i_{1,T_0})$. Finally by using the corrupted and computed values he/she calculates $K^{i+1}_{1,\mathrm{att}}$, $K^{i+1}_{2,\mathrm{att}}$ and $C^{i+1}_{T_0,\mathrm{att}}$ as follows:

$$\begin{aligned} K^{i+1}_{1,\mathrm{att}} &= \mathrm{Con}(K^i_{1,T_0}, n^{i+1}_{\mathrm{att}}) \oplus K^i_{2,T_0}, \\ K^{i+1}_{2,\mathrm{att}} &= \mathrm{Con}(K^i_{2,T_0}, B^{i+1}_{\mathrm{att}}) \oplus K^i_{1,T_0}, \\ C^{i+1}_{T_0,\mathrm{att}} &= \mathrm{Con}(\mathrm{Con}(B^{i+1}_{\mathrm{att}}, K^{i+1}_{1,\mathrm{att}}), \mathrm{Con}(K^{i+1}_{1,\mathrm{att}}, K^{i+1}_{2,\mathrm{att}} \\ &\oplus n^{i+1}_{\mathrm{att}})) \oplus ID_{T_0} \ . \end{aligned}$$

Guess phase The attacker $\mathcal{A}$ stops the game G and outputs a bit $b' \in \{0,\ 1\}$ as a guess for b. In order to determine $b' \in \{0,\ 1\}$, the attacker uses the calculated value $C^{i+1}_{T_0,\mathrm{att}}$ and the conversion function and computes $C^{i+1}_{LorR,T_0\mathrm{att}}$. Then he/she uses the following rule:

$$b' = \begin{cases} 0 & \text{if } C^{i+1}_{LorR,T_b} = C^{i+1}_{LorR,T_0\mathrm{att}} \\ 1 & \text{otherwise} \end{cases} .$$

Claim The SLAP protocol fails to provide forward untraceability and, in particular, for the constructed attacker $\mathcal{A}$, we have $\mathrm{Adv}^{\mathrm{UPriv}}_{\mathcal{A}}(\kappa) = \frac{1}{2}$.

Proof Since the secret value ID is fixed in all sessions and according to the structure of the SLAP protocol, following computations are straightforward:

$$\begin{aligned} C^{i+1}_{T_0,\mathrm{att}} &= \mathrm{Con}(\mathrm{Con}(B^{i+1}_{\mathrm{att}}, K^{i+1}_{1,\mathrm{att}}), \mathrm{Con}(K^{i+1}_{1,\mathrm{att}}, K^{i+1}_{2,\mathrm{att}} \\ &\quad \oplus n^{i+1}_{\mathrm{att}})) \oplus ID_{T_0} \\ &= \mathrm{Con}(\mathrm{Con}(B^{i+1}_{\mathrm{att}}, \mathrm{Con}(K^i_{1,T_0}, n^{i+1}_{\mathrm{att}}) \oplus K^i_{2,T_0}), \\ &\quad \mathrm{Con}(\mathrm{Con}(K^i_{1,T_0}, n^{i+1}_{\mathrm{att}}) \oplus K^i_{2,T_0}, \mathrm{Con}(K^i_{2,T_0}, \\ &\quad B^{i+1}_{\mathrm{att}}) \oplus K^i_{1,T_0} \oplus n^{i+1}_{\mathrm{att}})) \oplus ID_{T_0} \\ &= \mathrm{Con}(\mathrm{Con}(\mathrm{Con}(\mathrm{Rot}(K^i_{1,T_0}, n^{i+1}_{\mathrm{att}}), K^i_{1,T_0} \oplus \\ &\quad K^i_{2,T_0}) \oplus \mathrm{Rot}(\mathrm{Con}(K^i_{2,T_0}, K^i_{2,T_0} \oplus n^{i+1}_{\mathrm{att}}), K^i_{1,T_0}), \\ &\quad \mathrm{Con}(K^i_{1,T_0}, n^{i+1}_{\mathrm{att}}) \oplus K^i_{2,T_0}), \mathrm{Con}(\mathrm{Con}(K^i_{1,T_0}, \\ &\quad n^{i+1}_{\mathrm{att}}) \oplus K^i_{2,T_0}, \mathrm{Con}(K^i_{2,T_0}, \mathrm{Con}(\mathrm{Rot}(K^i_{1,T_0}, n^{i+1}_{\mathrm{att}}), \\ &\quad K^i_{1,T_0} \oplus K^i_{2,T_0}) \oplus \mathrm{Rot}(\mathrm{Con}(K^i_{2,T_0}, K^i_{2,T_0} \oplus n^{i+1}_{\mathrm{att}}), \\ &\quad K^i_{1,T_0})) \oplus K^i_{1,T_0} \oplus n^{i+1}_{\mathrm{att}})) \oplus ID_{T_0} \ . \end{aligned}$$

The first equation is derived by substituting the values for $K_{1,\mathrm{att}}^{i+1}$ and $K_{2,\mathrm{att}}^{i+1}$ and the second equation is coming from the relation $B_{\mathrm{att}}^{i+1} = \mathrm{Con}(\mathrm{Rot}(K_{1,T_0}^i, n_{\mathrm{att}}^{i+1}), K_{1,T_0}^i \oplus K_{2,T_0}^i) \oplus \mathrm{Rot}(\mathrm{Con}(\ K_{2,T_0}^i, K_{2,T_0}^i \oplus n_{\mathrm{att}}^{i+1}), K_{1,T_0}^i)$.

Now, by first substituting the value n_{att}^{i+1} and then $\alpha = \mathrm{Con}(K_{1,T_0}^i, K_{2,T_0}^i)$, we get:

$$\begin{aligned}
C_{T_0,\mathrm{att}}^{i+1} &= \mathrm{Con}(\mathrm{Con}(\mathrm{Con}(\mathrm{Rot}(K_{1,T_0}^i, A_{T_b}^{i+1} \oplus \alpha), K_{1,T_0}^i \oplus \\
&\quad K_{2,T_0}^i) \oplus \mathrm{Rot}(\mathrm{Con}(K_{2,T_0}^i, K_{2,T_0}^i \oplus A_{T_b}^{i+1} \oplus \alpha), \\
&\quad K_{1,T_0}^i), \mathrm{Con}(K_{1,T_0}^i, A_{T_b}^{i+1} \oplus \alpha) \oplus K_{2,T_0}^i), \mathrm{Con}(\\
&\quad \mathrm{Con}(K_{1,T_0}^i, A_{T_b}^{i+1} \oplus \alpha) \oplus K_{2,T_0}^i, \mathrm{Con}(K_{2,T_0}^i, \\
&\quad \mathrm{Con}(\mathrm{Rot}(K_{1,T_0}^i, A_{T_b}^{i+1} \oplus \alpha), K_{1,T_0}^i \oplus K_{2,T_0}^i) \oplus \\
&\quad \mathrm{Rot}(\mathrm{Con}(K_{2,T_0}^i, K_{2,T_0}^i \oplus A_{T_b}^{i+1} \oplus \alpha), K_{1,T_0}^i)) \oplus \\
&\quad K_{1,T_0}^i \oplus A_{T_b}^{i+1} \oplus \alpha)) \oplus ID_{T_0} \\
&= \mathrm{Con}(\mathrm{Con}(\mathrm{Con}(\mathrm{Rot}(K_{1,T_0}^i, A_{T_b}^{i+1} \oplus \mathrm{Con}(K_{1,T_0}^i, \\
&\quad K_{2,T_0}^i)), K_{1,T_0}^i \oplus K_{2,T_0}^i) \oplus \mathrm{Rot}(\mathrm{Con}(K_{2,T_0}^i, K_{2,T_0}^i \\
&\quad \oplus A_{T_b}^{i+1} \oplus \mathrm{Con}(K_{1,T_0}^i, K_{2,T_0}^i)), K_{1,T_0}^i), \mathrm{Con}(K_{1,T_0}^i, \\
&\quad A_{T_b}^{i+1} \oplus \mathrm{Con}(K_{1,T_0}^i, K_{2,T_0}^i)) \oplus K_{2,T_0}^i), \mathrm{Con}(\\
&\quad \mathrm{Con}(K_{1,T_0}^i, A_{T_b}^{i+1} \oplus \mathrm{Con}(K_{1,T_0}^i, K_{2,T_0}^i)) \oplus K_{2,T_0}^i, \\
&\quad \mathrm{Con}(K_{2,T_0}^i, \mathrm{Con}(\mathrm{Rot}(K_{1,T_0}^i, A_{T_b}^{i+1} \oplus \mathrm{Con}(K_{1,T_0}^i, \\
&\quad K_{2,T_0}^i)), K_{1,T_0}^i \oplus K_{2,T_0}^i) \oplus \mathrm{Rot}(\mathrm{Con}(K_{2,T_0}^i, \\
&\quad K_{2,T_0}^i \oplus A_{T_b}^{i+1} \oplus \mathrm{Con}(K_{1,T_0}^i, K_{2,T_0}^i)), K_{1,T_0}^i) \\
&\quad \oplus K_{1,T_0}^i \oplus A_{T_b}^{i+1} \oplus \mathrm{Con}(K_{1,T_0}^i, K_{2,T_0}^i))) \oplus ID_{T_0} .
\end{aligned}$$

Assuming $T_b = T_0$ and taking into account the following relations,

$$\begin{aligned}
n_{T_b}^{i+1} &= A_{T_b}^{i+1} \oplus \mathrm{Con}(K_{1,T_0}^i, K_{2,T_0}^i)\ , \\
B_{T_b}^{i+1} &= \mathrm{Con}(\mathrm{Rot}(K_{1,T_b}^i, n_{T_b}^{i+1}), K_{1,T_b}^i \oplus K_{2,T_b}^i) \oplus \\
&\quad \mathrm{Rot}(\mathrm{Con}(K_2^i, K_2^i \oplus n_{T_b}^{i+1}), K_1^i)\ , \\
K_{1,T_b}^{i+1} &= \mathrm{Con}(K_{1,T_b}^i, n_{T_b}^{i+1}) \oplus K_{2,T_b}^i\ , \\
K_{2,b}^{i+1} &= \mathrm{Con}(K_{2,T_b}^i, B_{T_b}^{i+1}) \oplus K_{1,T_b}^i\ ,
\end{aligned}$$

we reach our final simplification as follows:

$$\begin{aligned}
C_{T_0,\mathrm{att}}^{i+1} &= \mathrm{Con}(\mathrm{Con}(\mathrm{Con}(\mathrm{Rot}(K_{1,T_b}^i, n_{T_b}^{i+1}), K_{1,T_b}^i \oplus K_{2,T_b}^i) \\
&\quad \oplus \mathrm{Rot}(\mathrm{Con}(K_{2,T_b}^i, K_{2,T_b}^i \oplus n_{T_b}^{i+1}), K_{1,T_b}^i), \mathrm{Con}(\\
&\quad K_{1,T_b}^i, n_{T_b}^{i+1}) \oplus K_{2,T_b}^i), \mathrm{Con}(\mathrm{Con}(K_{1,T_b}^i, n_{T_b}^{i+1}) \\
&\quad \oplus K_{2,T_b}^i, \mathrm{Con}(K_{2,T_b}^i, \mathrm{Con}(\mathrm{Rot}(K_{1,T_b}^i, n_{T_b}^{i+1}), \\
&\quad K_{1,T_b}^i \oplus K_{2,T_b}^i) \oplus \mathrm{Rot}(\mathrm{Con}(K_{2,T_b}^i, K_{2,T_b}^i \oplus \\
&\quad n_{T_b}^{i+1})) \oplus K_{1,T_b}^i \oplus n_{T_b}^{i+1}), K_{1,T_b}^i) \oplus ID_{T_b} \\
&= \mathrm{Con}(\mathrm{Con}(B_{T_b}^{i+1}, K_{1,T_b}^{i+1}), \mathrm{Con}(K_{1,T_b}^{i+1}, K_{2,T_b}^{i+1} \oplus \\
&\quad n_{T_b}^{i+1})) \oplus ID_{T_b} \\
&= C_{T_b}^{i+1}\ .
\end{aligned}$$

Therefore, by performing the conversion function, we get the result that $C_{LorR,T_b}^{i+1} = C_{LorR,T_0,\mathrm{att}}^{i+1}$, i.e., $\Pr[b' = b] = 1$ which proves our claim.

6 Results

The section summarizes the results of the paper and presents a short review on the state of the art in the same family of RFID authentication protocols.

In Section 3, we studied GH protocol [25] and discovered some flaws in its structure for the first time. We proved that anonymity of GH protocol has some serious problems and it is breakable against traceability and forward traceability attacks. Our traceability attacks were presented in the *Ouafi-Phan* privacy model framework which is a well-known RFID privacy model. Then, in Section 4, we cryptanalyzed NZCL, another lightweight authentication protocol proposed by *Niu et al.* for RFID systems with mobile reader [26]. We observe that NZCL does not guarantee anonymity and security of RFID users completely and it has several vulnerabilities. In fact, in the NZCL protocol, an attacker not only is able to discover secret parameters of a particular tag (with maximum 2^L execution of an PRNG function, where L is the length of secret keys and typically chosen to be 16), but can also perform several independent and practical attacks including impersonation, DoS and forward traceability attacks. Finally, we analyzed SLAP, a novel and succinct protocol proposed by *Luo et al.* in [28]. We formally broke anonymity of SLAP's traceability and forward traceability. Table 2 summarizes our presented analysis on the three analyzed protocols (GH, NZCL, and SLAP [25,26,28]). However, in order to report the state-of-the-art results on similar RFID authentication protocols, current security status of some recent proposed RFID authentication protocols are also given.

Table 3 summarizes the cost of all presented attacks against the three analyzed protocols based on number of challenges with the target tag and the number of queries in the Ouafi-Phan formal privacy model or adhoc queries (or eavesdropping). For example, in section 3.3, we observed that in order to perform a for-

Table 2: Performance of studied protocols and some recently proposed ones

Protocols \ Attacks	A	B	C	D	E	F	in
Cho et al. [22]	✓	×	×	✓	×	✓	[76]
Safkhani et al. [21]	✓	✓	✓	✓	×	×	[31]
Improved Safkhani et al. [31]	✓	✓	✓	✓	✓	✓	—
Yoon [52]	✓	✓	✓	×	×	×	[7]
Improved Yoon [7]	✓	✓	✓	✓	✓	✓	—
Sun-Zhang [20]	✓	✓	✓	✓	✓	×	[7]
Improved Sun-Zhang [31]	✓	✓	✓	✓	✓	✓	—
GH [25]	✓	✓	✓	✓	×	×	Here
NZCL [26]	×	×	×	✓	✓	×	Here
SLAP [28]	✓	✓	✓	✓	×	×	Here

A: Secret Parameter Reveal B: DoS Attack C: Impersonation D: Backward Traceability Attack E: Traceability Attack F: Forward Traceability Attack

Table 3: The cost of presented attacks.

Attack \ Protocol	GH [25]	NZCL [26]	SLAP [28]
Traceability	2 C + 4 Q_F	-	2 C + 4 Q_F
Forward Traceability	2 C + 3 Q_F	2 C + 4 Q_F	2 C + 3 Q_F
Secret Parameters Reveal	-	1 C + 2 Q_A	-
DoS	-	2 C + 3 Q_A	-

C: One challenge with the target tag (by an attacker or a valid reader) Q_F: A query in Quafi-Phan formal privacy model Q_A: An ad-hoc execute query (eavesdropping).

ward traceability attack against GH protocol, an attacker $\mathcal{A}$ needs to send a $\textsc{Corrupt}(T_0, K')$ query, and a $\textsc{Test}(T_0, T_1, i)$ query in ith round, and additionally an $\textsc{Execute}(R, T_b, i{+}1)$ query in $(i{+}1)$th round; hence, the attack cost is $2\ C + 3\ \mathsf{Q}_F$.

Eventually, Table 4 compares the computational complexity of a tag, a reader and the back-end server for each tag in all the analyzed protocols. As it can be seen, however for all parties, NZCL is more lightweight (it needs 2 PRF + 1 RNG for computation of each tag, and 4 PRF + 1 RNG for authentication each tag in the back-end server) and GH is the heaviest one (it needs 6 H+1 RNG for computation of each tag, and 7 H for authentication each tag in the back-end server) among the analyzed protocols. However, our analysis shows that they need to be refined before using in practical systems which we leave as a open problem for future research.

Table 4: A comparison of analyzed protocols from computational complexity point of view.

Party \ Protocol	GH [25]	NZCL [26]	SLAP [28]
Tag's Complexity	6 H + 1 RNG	2 PRF + 1 RNG	6 Con + 1 RNG
Reader's Complexity	2 H + 1 RNG	1 RNG	13 Con + 2 Rot
Server's Complexity	7 H	4 PRF + 1 RNG	

H: Hash function RNG: Random Number Generator Con: Conversion function Rot: Cyclic left rotation based on Hamming weight.

7 Conclusion

In this study, we have cryptanalyzed GH [25], NZCL [26] and SLAP [28], three recently proposed RFID lightweight authentication protocols which were claimed to provide strong privacy for end-users. We have shown that all the aforementioned protocols have some security and privacy drawbacks which have made them insecure against several practical attacks. We proved that GH and SLAP both are insecure against *traceability* and *forward traceability* attacks and NZCL is vulnerable to *secret parameter reveal*, *impersonation*, *DoS*, and *forward traceability* attacks.

Acknowledgements Two first authors were supported by institutional research funding IUT2-1 of the Estonian Ministry of Education and Research. The third author has been supported by Iranian National Science Foundation (INSF) under contract No. 92027548 and Sharif Industrial Relation Office (SIRO) under Grant No. G931223.

References

1. W. Xie, L. Xie, C. Zhang, Q. Wang, J. Xu, Q. Zhang, and C. Tang, "RFID seeking: Finding a lost tag rather than only detecting its missing," *Journal of Network and Computer Applications*, vol. 42, pp. 135–142, 2014.

2. M. Tajima, "Strategic value of RFID in supply chain management," *Journal of purchasing and supply management*, vol. 13, no. 4, pp. 261–273, 2007.
3. N. Van Deursen, W. J. Buchanan, and A. Duff, "Monitoring information security risks within health care," *computers & security*, vol. 37, pp. 31–45, 2013.
4. H. Gross, E. Wenger, H. Martin, and M. Hutter, "Pioneer-prototype for the internet of things based on an extendable EPC gen2 RFID tag," in *International Workshop on Radio Frequency Identification: Security and Privacy Issues*, pp. 54–73, Springer, 2014.
5. A. Galins, P. Beinarovics, A. Laizans, A. Jakusenoks, *et al.*, "RFID application for electric car identification at charging station," in *Engineering for Rural Development: Proceedings of the 15th International scientific conference.*, pp. 25–27, 2016.
6. M. S. Farash, M. Turkanović, S. Kumari, and M. Hölbl, "An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the internet of things environment," *Ad Hoc Networks*, vol. 36, pp. 152–176, 2016.
7. K. Baghery, B. Abdolmaleki, B. Akhbari, and M. R. Aref, "Enhancing privacy of recent authentication schemes for low-cost RFID systems," *The ISC International Journal of Information Security*, vol. 7, no. 2, pp. 135–149, 2016.
8. W. S. Suh, E. J. Yoon, and S. Piramuthu, "RFID-based attack scenarios in retailing, healthcare and sports," *Journal of Information Privacy and Security*, vol. 9, no. 3, pp. 4–17, 2013.
9. H. Jannati, "Analysis of relay, terrorist fraud and distance fraud attacks on RFID systems," *International Journal of Critical Infrastructure Protection*, vol. 11, pp. 51–61, 2015.
10. S. Amendola, R. Lodato, S. Manzari, C. Occhiuzzi, and G. Marrocco, "RFID technology for iot-based personal healthcare in smart spaces," *IEEE Internet of things journal*, vol. 1, no. 2, pp. 144–152, 2014.
11. B. Khoo, "RFID as an enabler of the internet of things: Issues of security and privacy," in *Internet of Things (iThings/CPSCom), 2011 International Conference on and 4th International Conference on Cyber, Physical and Social Computing*, pp. 709–712, IEEE, 2011.
12. M. Bolic, M. Rostamian, and P. M. Djuric, "Proximity detection with RFID: A step toward the internet of things," *IEEE Pervasive Computing*, vol. 14, no. 2, pp. 70–76, 2015.
13. I. Memon, Q. A. Arain, H. Memon, and F. A. Mangi, "Efficient user based authentication protocol for location based services discovery over road networks," *Wireless Personal Communications*, pp. 1–20.
14. I. Memon, I. Hussain, R. Akhtar, and G. Chen, "Enhanced privacy and authentication: An efficient and secure anonymous communication for location based service using asymmetric cryptography scheme," *Wireless Personal Communications*, vol. 84, no. 2, pp. 1487–1508, 2015.
15. L. Da Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.
16. E. Welbourne, L. Battle, G. Cole, K. Gould, K. Rector, S. Raymer, M. Balazinska, and G. Borriello, "Building the internet of things using RFID: the RFID ecosystem experience," *IEEE Internet Computing*, vol. 13, no. 3, pp. 48–55, 2009.
17. Y. Shifeng, F. Chungui, H. Yuanyuan, and Z. Shiping, "Application of iot in agriculture," *Journal of Agricultural Mechanization Research*, vol. 7, pp. 190–193, 2011.
18. J. Wang, D. Ni, and K. Li, "RFID-based vehicle positioning and its applications in connected vehicles," *Sensors*, vol. 14, no. 3, pp. 4225–4238, 2014.
19. T. Hayajneh, B. J. Mohd, M. Imran, G. Almashaqbeh, and A. V. Vasilakos, "Secure authentication for remote patient monitoring with wireless medical sensor networks," *Sensors*, vol. 16, no. 4, p. 424, 2016.
20. D.-Z. Sun and J.-D. Zhong, "A hash-based RFID security protocol for strong privacy protection," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 4, pp. 1246–1252, 2012.
21. M. Safkhani, N. Bagheri, and M. Naderi, "On the designing of a tamper resistant prescription rfid access control system," *Journal of medical systems*, vol. 36, no. 6, pp. 3995–4004, 2012.
22. J.-S. Cho, Y.-S. Jeong, and S. O. Park, "Consideration on the brute-force attack cost and retrieval cost: A hash-based radio-frequency identification (RFID) tag mutual authentication protocol," *Computers & Mathematics with Applications*, vol. 69, no. 1, pp. 58–65, 2015.
23. M. S. Farash, "Cryptanalysis and improvement of an efficient mutual authentication RFID scheme based on elliptic curve cryptography," *The Journal of Supercomputing*, vol. 70, no. 2, pp. 987–1001, 2014.
24. C.-L. Chen and Y.-Y. Deng, "Conformation of EPC class 1 generation 2 standards RFID system with mutual authentication and privacy protection," *Engineering Applications of Artificial Intelligence*, vol. 22, no. 8, pp. 1284–1291, 2009.
25. P. Gope and T. Hwang, "A realistic lightweight authentication protocol preserving strong anonymity for securing RFID system," *Computers & Security*, vol. 55, pp. 271–280, 2015.
26. B. Niu, X. Zhu, H. Chi, and H. Li, "Privacy and authentication protocol for mobile RFID systems," *Wireless personal communications*, vol. 77, no. 3, pp. 1713–1731, 2014.
27. K. Baghery, B. Abdolmaleki, B. Akhbari, and M. R. Aref, "Privacy analysis and improvements of two recent RFID authentication protocols," in *Information Security and*

Cryptology (ISCISC), 2014 11th International ISC Conference on, pp. 137–142, IEEE, 2014.

28. H. Luo, G. Wen, J. Su, and Z. Huang, "SLAP: Succinct and lightweight authentication protocol for low-cost RFID system," *Wireless Networks*, pp. 1–10, 2016.
29. S. M. Alavi, K. Baghery, and B. Abdolmaleki, "Security and privacy flaws in a recent authentication protocol for EPC C1 G2 RFID tags," *Advances in Computer Science: An International Journal*, vol. 3, no. 5, pp. 44–52, 2014.
30. D. He and S. Zeadally, "An analysis of RFID authentication schemes for internet of things in healthcare environment using elliptic curve cryptography," *IEEE internet of things journal*, vol. 2, no. 1, pp. 72–83, 2015.
31. B. Abdolmaleki, K. Baghery, S. Khazaei, and M. R. Aref, "Game-based privacy analysis of RFID security schemes for confident authentication in IoT," *Wireless Personal Communications*, pp. 1–24, 2017.
32. S. M. Alavi, K. Baghery, B. Abdolmaleki, and M. R. Aref, "Traceability analysis of recent RFID authentication protocols," *Wireless Personal Communications*, vol. 83, no. 3, pp. 1663–1682, 2015.
33. M. Akgün, A. O. Bayrak, and M. U. Çaglayan, "Attacks and improvements to chaotic map-based RFID authentication protocol," *Security and Communication Networks*, vol. 8, no. 18, pp. 4028–4040, 2015.
34. B. Abdolmaleki, K. Baghery, B. Akhbari, S. M. Alavi, and M. R. Aref, "Securing key exchange and key agreement security schemes for rfid passive tags," in *Electrical Engineering (ICEE), 2016 24th Iranian Conference on*, pp. 1475–1480, IEEE, 2016.
35. F. Moradi, H. Mala, B. T. Ladani, and F. Moradi, "Security analysis of an epc class-1 generation-2 compliant rfid authentication protocol," *Journal of Computing and Security*, vol. 3, no. 3, 2018.
36. N. J. Hopper and M. Blum, "Secure human identification protocols," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 52–66, Springer, 2001.
37. A. Juels and S. A. Weis, "Authenticating pervasive devices with human protocols," in *Annual International Cryptology Conference*, pp. 293–308, Springer, 2005.
38. J. Bringer, H. Chabanne, and E. Dottax, "Hbˆ+ˆ+: a lightweight authentication protocol secure against some attacks," in *Second International Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing (SecPerU'06)*, pp. 28–33, IEEE, 2006.
39. S. Piramuthu, "Hb and related lightweight authentication protocols for secure RFID tag/reader authentication title," *CollECTeR Europe 2006*, p. 239, 2006.
40. P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estévez-Tapiador, and A. Ribagorda, "Lmap: A real lightweight mutual authentication protocol for low-cost RFID tags," in *Workshop on RFID security*, pp. 12–14, 2006.
41. P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda, "M2ap: A minimalist mutual-authentication protocol for low-cost RFID tags," in *International Conference on Ubiquitous Intelligence and Computing*, pp. 912–923, Springer, 2006.
42. 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 Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pp. 352–361, Springer, 2006.
43. T. Li and G. Wang, "Security analysis of two ultra-lightweight RFID authentication protocols," in *IFIP International Information Security Conference*, pp. 109–120, Springer, 2007.
44. T. Li and R. Deng, "Vulnerability analysis of emap-an efficient RFID mutual authentication protocol," in *Availability, Reliability and Security, 2007. ARES 2007. The Second International Conference on*, pp. 238–245, IEEE, 2007.
45. H.-Y. Chien, "Sasi: A new ultralightweight RFID authentication protocol providing strong authentication and strong integrity," *IEEE Transactions on Dependable and Secure Computing*, vol. 4, no. 4, pp. 337–340, 2007.
46. R. C. Phan, "Cryptanalysis of a new ultralightweight RFID authentication protocol-sasi," *IEEE Transactions on Dependable and Secure Computing*, vol. 6, no. 4, p. 316, 2009.
47. G. Avoine, X. Carpent, and B. Martin, "Strong authentication and strong integrity (sasi) is not that strong," in *International Workshop on Radio Frequency Identification: Security and Privacy Issues*, pp. 50–64, Springer, 2010.
48. G. Avoine, X. Carpent, and B. Martin, "Privacy-friendly synchronized ultralightweight authentication protocols in the storm," *Journal of Network and Computer Applications*, vol. 35, no. 2, pp. 826–843, 2012.
49. D. N. Duc, H. Lee, and K. Kim, "Enhancing security of epcglobal Gen-2 RFID against traceability and cloning," *Auto-ID Labs Information and Communication University, White Paper*, 2006.
50. S. Karthikeyan and M. Nesterenko, "RFID security without extensive cryptography," in *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, pp. 63–67, ACM, 2005.
51. H.-Y. Chien and C.-H. Chen, "Mutual authentication protocol for RFID conforming to EPC class 1 generation 2 standards," *Computer Standards & Interfaces*, vol. 29, no. 2, pp. 254–259, 2007.
52. E.-J. Yoon, "Improvement of the securing RFID systems conforming to EPC class 1 generation 2 standard," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1589–1594, 2012.

53. J. Ha, S. Moon, J. Zhou, and J. Ha, "A new formal proof model for RFID location privacy," in *European Symposium on Research in Computer Security*, pp. 267–281, Springer, 2008.
54. S. W. Jung and S. Jung, "Hmac-based RFID authentication protocol with minimal retrieval at server," in *The Fifth International Conference on Evolving Internet*, pp. 52–55, 2013.
55. Y.-Y. Chen, D.-C. Huang, M.-L. Tsai, and J.-K. Jan, "A design of tamper resistant prescription RFID access control system," *Journal of medical systems*, vol. 36, no. 5, pp. 2795–2801, 2012.
56. M. Safkhani, N. Bagheri, and M. Naderi, "On the designing of a tamper resistant prescription rfid access control system," *Journal of medical systems*, vol. 36, no. 6, pp. 3995–4004, 2012.
57. B.-H. Liu, N.-T. Nguyen, V.-T. Pham, and Y.-H. Yeh, "A maximum-weight-independent-set-based algorithm for reader-coverage collision avoidance arrangement in rfid networks," *IEEE Sensors Journal*, vol. 16, no. 5, pp. 1342–1350, 2016.
58. F. Rahman, M. E. Hoque, and S. I. Ahamed, "Anonpri: A secure anonymous private authentication protocol for rfid systems," *Information Sciences*, vol. 379, pp. 195–210, 2017.
59. F. Rahman, M. Z. A. Bhuiyan, and S. I. Ahamed, "A privacy preserving framework for rfid based healthcare systems," *Future Generation Computer Systems*, vol. 72, pp. 339–352, 2017.
60. N.-T. Nguyen, B.-H. Liu, and V.-T. Pham, "A dynamic-range-based algorithm for reader-tag collision avoidance deployment in rfid networks," in *Electronics, Information, and Communications (ICEIC), 2016 International Conference on*, pp. 1–4, IEEE, 2016.
61. B. J. Mohd, T. Hayajneh, Z. A. Khalaf, A. Yousef, and K. Mustafa, "Modeling and optimization of the lightweight hight block cipher design with fpga implementation," *Security and Communication Networks*, vol. 9, no. 13, pp. 2200–2216, 2016.
62. B. J. Mohd, T. Hayajneh, and A. V. Vasilakos, "A survey on lightweight block ciphers for low-resource devices: Comparative study and open issues," *Journal of Network and Computer Applications*, vol. 58, pp. 73–93, 2015.
63. K. Ouafi and R. C.-W. Phan, "Privacy of recent RFID authentication protocols," in *Information Security Practice and Experience*, pp. 263–277, Springer, 2008.
64. B. Abdolmaleki, K. Baghery, B. Akhbari, and M. R. Aref, "Attacks and improvements on two new-found RFID authentication protocols," in *Telecommunications (IST), 2014 7th International Symposium on*, pp. 895–900, IEEE, 2014.
65. A. Juels, "RFID security and privacy: A research survey," *IEEE journal on selected areas in communications*, vol. 24, no. 2, pp. 381–394, 2006.
66. S. Vaudenay, "On privacy models for RFID," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 68–87, Springer, 2007.
67. I. Coisel and T. Martin, "Untangling RFID privacy models," *Journal of Computer Networks and Communications*, vol. 2013, 2013.
68. G. Avoine, "Adversarial model for radio frequency identification.," *IACR Cryptology ePrint Archive*, vol. 2005, p. 49, 2005.
69. A. Juels and S. A. Weis, "Defining strong privacy for RFID," *ACM Transactions on Information and System Security (TISSEC)*, vol. 13, no. 1, p. 7, 2009.
70. R. H. Deng, Y. Li, M. Yung, and Y. Zhao, "A new framework for RFID privacy," in *European Symposium on Research in Computer Security*, pp. 1–18, Springer, 2010.
71. J. Hermans, A. Pashalidis, F. Vercauteren, and B. Preneel, "A new RFID privacy model," in *European Symposium on Research in Computer Security*, pp. 568–587, Springer, 2011.
72. M. H. Habibi, M. R. Aref, and D. Ma, "Addressing flaws in RFID authentication protocols," in *International Conference on Cryptology in India*, pp. 216–235, Springer, 2011.
73. R. C.-W. Phan, J. Wu, K. Ouafi, and D. R. Stinson, "Privacy analysis of forward and backward untraceable RFID authentication schemes," *Wireless Personal Communications*, vol. 61, no. 1, pp. 69–81, 2011.
74. M. R. Alagheband and M. R. Aref, "Unified privacy analysis of new-found RFID authentication protocols," *Security and Communication Networks*, vol. 6, no. 8, pp. 999–1009, 2013.
75. S. Wang, S. Liu, and D. Chen, "Security analysis and improvement on two RFID authentication protocols," *Wireless Personal Communications*, vol. 82, no. 1, pp. 21–33, 2015.
76. M. Safkhani, P. Peris-Lopez, J. C. Hernandez-Castro, and N. Bagheri, "Cryptanalysis of the cho et al. protocol: A hash-based RFID tag mutual authentication protocol," *Journal of Computational and Applied Mathematics*, vol. 259, pp. 571–577, 2014.