Are RNGs Achilles' heel of RFID Security and Privacy Protocols ?

Atakan Arslan[a,*], Süleyman Kardaş[b], Sultan Aldırmaz[a], Sarp Ertürk[a]

[a] *Kocaeli University, Department of Electronics and Communication, Kocaeli, Turkey*
[b] *Batman University, Faculty of Engineering and Architecture, Batman, Turkey*

Abstract

Security and privacy concerns have been growing with the increased usage of the RFID technology in our daily lives. To mitigate these issues, numerous privacy-friendly authentication protocols have been published in the last decade. Random number generators (RNGs) are commonly used in RFID tags to provide security and privacy of RFID protocols. RNGs might be weak spot of a protocol scheme and misusing of RNGs causes security and privacy problems. However, having a secure RNG with large entropy might be a trade-off between security and cost for low-cost RFID tags. Furthermore, a RNG used in RFID tag may not work properly in time. Therefore, we claim that vulnerability of using a RNG may deeply influence the security and privacy level of the system. To the best of our knowledge, this concern has not been considered in RFID literature. Motivated by this need, in this study, we first revisit Vaudenay's privacy model which combines the early models and presents a new mature and elegant privacy model with different adversary classes. Then, we enhance the model by introducing a new oracle, which allows analyzing the usage of RNGs in RFID protocols. We also analyze a couple of proposed protocols under our improved model.

Keywords: RFID, Protocol, Privacy, Security, RNG

1. Introduction

Radio **F**requency **ID**entification (RFID) has become one of the most emerging wireless technologies in order to identify and authenticate objects, animals and people in recent years. RFID is also one of the most likely technologies to promote the Internet of things (IoT) paradigm and is proliferated in many real-life applications such as access control, supply chain, hospital care system, automatic toll collection, payment systems, e-passport, vicinity/proximity cards

*Corresponding author. E-mail: atknarsln@gmail.com
Email addresses: atknarsln@gmail.com (Atakan Arslan), skardas@gmail.com (Süleyman Kardaş), sultan.aldirmaz@kocaeli.edu.tr (Sultan Aldırmaz), sertur@kocaeli.edu.tr (Sarp Ertürk)

etc. It is considered that near-field communication (NFC) technology in smart phones is a new up-to-the-minute opportunities for RFID technology and we are on the doorstep of new RFID era [1, 2].

A simple RFID system consists of a tag *(transponder)*, a reader *(interrogator)* and a back-end server. A tag basically has a microchip which stores data and an antenna used to transmit and receive messages with converting electromagnetic waves. Generally, it is considered that a back-end server is separated from a RFID reader and a RFID reader acts as a mediator between tags and server for the communication. A back-end server keeps the all information (secret keys, data etc.) about tags. Furthermore, RFID tags can be categorized as active, passive and semi-passive. Passive tags do not have own power source and energize their integrated circuit (IC) by using the waves transmitted by the reader. Moreover, tags can also divided into four groups with respect to the operating frequency that depends on the availability of frequency bands and the regulations: Low frequency (LF, 125-134.2 kHz and 140-148.5 kHz), high frequency (HF, ISM band at 13.56 MHz), ultra high frequency (UHF, 860-960 MHz) and microwave (>2.45 GHz) [2]. Passive low-cost RFID tags with smaller sizes are highly preferred in many applications and this desire brings some computation, energy and space restrictions on the tag. The range of the production price of the tags is around \$0.05 - \$0.10 and the cost pressure is quite dominant on hardware capabilities [3].

Security and privacy concerns arise since a tag communicates with a reader over an insecure wireless channel. Tag impersonating, tracking (forward and backward tracing), eavesdropping, replay, man-in-the-middle and denial of service (DoS) attacks can be performed by an attacker with using the messages transmitted in the air. These issues can be easily overcome with help of using conventional cryptography. However, the standard cryptographic algorithms cannot be implemented in the constrained tags [2, 3, 4, 5, 6]. The limited capabilities of RFID tags deepen security and privacy issues. Therefore, the designing secure and private lightweight authentication protocols has been a challenging and an important topic in RFID literature.

Over the past few years, numerous lightweight authentication protocols have been proposed so as to mitigate security and privacy concerns for RFID systems [7]. The most of them claimed that they were impregnable against to every type with providing different RFID system properties such as scalable identification, tag ownership transfer, mutual authentication, being robust against noisy environments, reader corruption resiliency etc. Unfortunately, many of them are failed to satisfies the claimed security and privacy concerns. [7, 8, 5, 9].

On the other hand, privacy models have been presented to systematically analyze the security and privacy of the proposed authentication protocols. Such an evaluation is theoretically accomplished based on those models to examine the security, anonymity and untraceability properties before using an RFID protocol in real-life systems. Recently, several models have been proposed to formalize security and privacy in the context of RFID system [10, 11, 12, 13, 14, 15, 16, 17, 18]. A privacy model should be detailed, attentive and flexible not to overlook the realities of the practical RFID systems. Although it has

been considered that Vaudenay's model [12] is one of the most evolved and well defined privacy model, some papers have been published to ameliorate the Vaudenay's model [14, 17, 18, 19]. These publications to the best of our knowledge have been claimed that their improvement fulfills the missing of the model but the privacy model has still fractures. In our opinion, designing a new, appropriate, complete and flexible security and privacy model with considering the various abilities of an adversary is an essential need. Besides, we have noticed that Vaudenay's model has not chewed on the misuse of random number generators and this is a new and different adversary ability especially for real-word scenarios.

Designers generally build the security and privacy of their protocols on the usage of Random Number Generator (RNG) which is one of the most common primitive cryptographic functions. Although RNGs are computationally secure, misusing them in the design causes serious weaknesses. More importantly, today many proposed RNGs that are asserted secure might be broken or become weaker in near future. If RFID protocol designers delude themselves into thinking indiscriminately using RNGs has no effect on security and privacy, they might be disappointed. From the point of this view, we claim that RNGs can be the weakest point in a RFID protocol. In this context, we extend the Vaudenay's model and define a new random oracle. In order to show our model works, we study two authentication protocols published in [20, 21]. These protocols use RNGs to hide their secrets on tag side by only XOR-ing. We show that an adversary is able to attack the protocols and recover the secrets because of the fact that entropy of the RNGs is assumed insufficient. Hence, we advice to the designers that RNGs should be only utilized to increase the security and privacy level of RFID protocols

Outline of the paper. The paper is structured as follows. In Section 2, related work is surveyed. In Section 3, our extended modification of Vaudenay's model is presented. In Section 4, the security and privacy of some proposed protocols are analyzed based on the modified model. We conclude the paper in Section 5 with a brief conclusion and future work.

2. Related Work

In this section, we will consider on several topics to highlight some other literature works such as lightweight protocols, privacy models, RNGs, computation capabilities and other RNG weaknesses.

2.1. Lightweight Protocols

Several lightweight and ultra-lightweight protocols for low-cost RFID tags have been published in the literature in order to obviate the security and privacy concerns [7]. In the design of these protocols, non-standard cryptographic functions and some basic simple operations can only be used on the tag side because of the aforementioned reasons that makes tags cheaper such as smaller area, lower energy and timing accuracy. Moreover, ultra-lightweight authentication protocols use simple operations (XOR, AND, OR), modular addition or

rotation etc. Some of the famous ones are SASI [22], LMAP [23], M2AP [24], EMAP [25] and Gossomar [26]. On the other hand, lightweight protocols use the same operations used by ultra-lightweight ones and RNGs, Cyclic Redundancy Check (CRC) and hash-functions. A few known protocols can be visited in [27, 28, 29]. However, the restrictions mentioned above greatly limits aptitudes of RFID tags and causes security and privacy vulnerabilities. In 2013, Avoine et al. [30] have evaluated, compared the well-known lightweight protocols and indicate the security and privacy weaknesses. Bilal has also most recently addressed the security and privacy issues in low-cost RFID systems in this PhD thesis [2].

2.2. Privacy Models

Privacy models are presented to be a base for analyzing the security and privacy of the authentication protocols in a methodologically manner. For this purpose, the privacy models formally define some properties such as RFID schemes, security and privacy prerequisites of the schemes and abilities of an adversary. In this context, Avoine et al. firstly published a framework to formalize privacy in RFID protocols in 2005 [31]. Avoine also extended the previous model in his thesis [10]. Then, Juels and Weis modify Avoine's model by adding side channel information attribute [11]. The following articles can be visited to see different model definitions [32, 33]. Although, there are many other attempts to design useful, proper and complete privacy model to represent and analyze RFID systems, the models do not consider or miss some important adversary properties (corruption, using side channel information etc.) and they do not appropriately modeling an RFID scheme: authentication, identification, protocol execution etc. However, in 2007, Vaudenay proposed a well-designed and complete privacy model in [12] and it will have been quite popular among many protocol designers. In time, some researches ameliorate the Vaudenay's model [17, 14, 18, 19].

2.3. RNGs

There are two types of random number generators: Pseudo-Random Number Generator (PRNG) and Truly Random Number Generator (TRNG). PRNG also known as a deterministic random number generator (DRNG) is an algorithm for generating random numbers with provided an initial value called a seed. The output of the PRNG is also called a pseudo-random bit sequence. The length of the output of a PRNG is much greater than the length of the seed. In addition to this, the output of a PRNG seems to be random because it has to be statistically indistinguishable from random values and also it is unpredictable when its seed is not known. Besides PRNG, True Random Number Generator (TRNG) is another algorithm that generates random numbers from a natural sources of randomness. Two general conditions are required from the security perceptive for a pseudorandom random generator: (i) the output of a PRNG should be statistically indistinguishable from truly random sequences, (ii) the next of the sequence should be unpredictable to an adversary with limited computational

resources. Theoretically, it can be predictable with negligible probability, 2^{-80}. In fact, the minimum security requirement is the length of the random seed has to be sufficiently large (s-bit) to be infeasible for the adversary to search over 2^s space. Sometimes s is called the security parameter.

It is impossible to prove that an output of a RNG is random but there are various statistical tests that measure the quality of a RNG. This is performed by taking a sample output sequence and apply the tests. The tests are probabilistic so they determine that whether the sample looks like a truly random sequence or not as a result. If the generator fails, the output is non-random. On the other hand, if a RNG passes all the test, it is not rejected as being non-random. The five basic tests of them are (i) frequency test (mono bit test), (ii) serial test (two-bit test), (iii) poker test, (iv) runs test, (v) auto-correlation test [34]. We do not want to digress so the following reference can be visited for detailed information about tests, generators, algorithms, definitions [34]. Moreover, some institutes, research centers, government agencies or organizations specify definite criteria to control the randomness of RNGs. For instance, the German Federal Office for Information Security has established several procedures for quality of the generators [35].

Extremely restricted computational, storage, energy and communication abilities of low-cost passive RFID tags are challenges to design a real lightweight RFID protocols. Unlike other RFID protocols, lightweight protocols only need the simplest bitwise operations (XOR, AND, OR, rotation, permutation, etc.), RNG and CRC etc. The usage of RNGs has become the key function in most private and secure lightweight RFID protocols for low cost RFID tags. Low cost RFID tags have approximately 5K-10K gates and their 0.4K-4K gates can be dedicated to security operations [36]. Furthermore, designers are also restricted with the time that is put out by a tag while generating random number because RFID readers should be able to read a bunch of tags in a certain amount of time.

Many publications have been proposed to design and use RNGs in low cost RFID tags. Melia-Segui et al. present a lightweight PRNG design for low-cost passive RFID tags, called J3Gen in 2013 [37]. J3Gen is based on a LFSR (Linear Feedback Shift Register) configured with multiple feedback polynomials that are changed during the generation of sequences by a physical source. They determined their most efficient J3Gen design that has 32-bit LFSR output with 16-bit feedback polynomials requires around 1.2K logic gate equivalence (GE). Furthermore, Peinado et al. [38] analyze J3Gen and they claim that there are two cryptanalytic attacks on J3Gen. In March 2015, Garcia-Alfaro et al. [39] show that Peinado et al.'s assumptions are incorrect and their attack against J3Gen is not valid. At this point, although Garcia-Alfaro et al. fend off the attack, the literature is waiting for the objections for J3Gen PRNG.

Peris et al. proposed a PRNG, named LAMED, for low cost RFID tags compliant with EPC C1G2 standard in 2009 [36]. They claim that LAMED successfully passes several randomness tests. LAMED requires roughly 1.6K gates and 1.9 ms to generate a 32-bit random number.

Melia-Segui et al. [40] present a practical attack on a weak PRNG proposed

by Che et al. [41] to design for EPC Gen2 tags. Che et al. proposed a LFSR based PRNG with the combination of thermal noise signal modulation. Melia-Segui et al. obtain the feedback polynomial function of the LFSR so they can predict its generated sequences. They show that an adversary can reach the PRNG configuration with a confidence of 42% by only eavesdropping 128 bits of PRNG data.

In 2008, Garcia et al. [42] has shown that the PRNG used in the MIFARE Classic chip has vulnerabilities.

In 2014, Armknecht et al. [3] have pointed out that ensuring a sufficient level of entropy for RNGs is still a difficult task. They said that different experts from industry who provides them information, all agrees that generating more than 128 true random bits per authentication on an RFID tag in the price range of $0.05-$0.10 seems currently improbable.

EPC C1G2 (Class-1 Gen-2) RFID standard was proposed and adopted by EPCglobal in 2004. In 2006, it was published as an amendment to ISO 18000-6 standard for low-cost lightweight UHF RFID tags. The new version of standards have been recently ratified in 2013 with some optional cryptographic properties [27, 43].

According to the standards, a tag generates 16-bit pseudo-random numbers (RN16) using the RNG. The RNG shall meet three randomness criteria: probability of a single RN16, probability of simultaneously identical sequences and probability of predicting an RN16. Although these requirements may be more stringent, brute-force attack can be applied to reveal the random numbers because lightweight low-cost RFID tags are able to use $32 - bit$ output of PRNG which is a weakness. An adversary eavesdrops the messages between reader and RFID tag, then brute-force attack or time-memory trade-off attack can be occurred to reach the secrets of a victim tag.

2.4. Computation Capabilities

Hashcat is the well-known fastest password recovery tool[44]. Versions are available for Linux, OSX, and Windows. It also comes in two variants: CPU-based (Hashcat password recovery tool) or GPU-based (oclHashcat, accelerated tool). oclHashcat is a GPU-based multi-hash cracker using a brute-force attack (implemented as mask attack), combinator attack, dictionary attack, hybrid attack, mask attack, and rule-based attack.

The performance of oclHashcat in different operating systems on MD5 and SHA1 is depicted in the table [44]. It means that PC3 can do 135232 Mh/s against MD5, which approximately means 0.135 billion tries per second. Hence, if the same computer is used for exhaustive search, less than 32 ms will be elapsed to find the result matched to the output of the 32-bit PRNG.

Hash Type	PC1[1]	PC2[2]	PC3[3]	PC4[4]
MD5	8581 Mh/s	2753 Mh/s	135232 Mh/s	92672 Mh/s
SHA1	3037 Mh/s	655 Mh/s	42408 Mh/s	31552 Mh/s
SHA256	1122 Mh/s	355 Mh/s	16904 Mh/s	12288 Mh/s
SHA512	414 Mh/s	104 Mh/s	5240 Mh/s	4552 Mh/s

2.5. Other RNG Weaknesses

RNGs are implemented by electronic circuits and their randomness quality might be affected by various factors such as seed entropy, aging, environmental effects (temperature, humidity, pressure, vibration, electromagnetic field, chemicals, etc.). As a result, biased RNGs cause irretrievable weaknesses.

Bayon et al. [45] demonstrate a practical attack ring oscillator (RO) based TRNGs by injecting an EM signal and they also mention the previous work about another practical assault to RO based TRNGs by injecting a sine wave signal onto the power pad of the device. Both attacks show that it is possible dynamically control the bias of the TRNG output.

In [34], the authors claimed that randomness and size of key generation affects to eliminate the advantages of adversaries. Then, they gave the following example that Data Encryption Standart (DES) encryption algorithm has 2^{56} key space size. When a secret key is selected by using a TRNG, an adversary has to try averagely 2^{55} possible searches to find the correct key. On the other hand, if the encryption key was selected by using 16-bit random secret and expanding it into a 56-bit key by using the well-known function, the adversary needs to try averagely 2^{15} possible keys to find the correct one.

In [46], the authors present a detailed survey paper about random number generators. They compare different type of PRNGs and TRNGs. They also criticize about real randomness, theoretic and practical RNGs approaches. They say that most researchers chose the minimum-action strategy: design a TRNG, obtain at least one random number sequence that passes chosen set of randomness tests and publish. However, it does not mean that those TRNGs have a really good randomness quality because small variations in hardware can weaken them. Hence, a theoretical design cannot proceed towards a product without a detailed investigation of hardware and without randomness proof. Furthermore, they reference that Barak, Shaltiel and Tomer proposed a extractor functions to make RNGs robust against to aging, temperature changes, etc. Moreover, they present a couple of weak RNGs because of hardware imperfections.

[1]PC1: Windows 7, 32 bit Catalyst 14.9 1x AMD hd7970 1000mhz core clock oclHashcat v1.35

[2]PC2: Windows 7, 64 bit ForceWare 347.52 1x NVidia gtx580 stock core clock oclHashcat v1.35

[3]PC3: Ubuntu 14.04, 64 bit ForceWare 346.29 8x NVidia Titan Xstock core clockoclHashcat v1.36

[4]PC4: Ubuntu 14.04, 64 bit Catalyst 14.9 8x AMD R9 290X stock core clock oclHashcat v1.35

3. The Modified Vaudenay Privacy Model

In this section, we introduce our modified version of the well-known Vaudenay Privacy Model [12] in order to analyse security and privacy level of RFID schemes. After that, we present the abilities of an adversary against to this model.

An RFID system is basically composed of three entities: a tag $\mathcal{T}$, a reader $\mathcal{R}$ and a back-end system/database $\mathcal{DB}$. A passive tag $\mathcal{T}$ is interrogated by a reader $\mathcal{R}$ and the reader identifies/authenticates $\mathcal{T}$ with using a unique identifier of the tag ID (in this article it is sometimes denoted $ID_{\mathcal{T}}$ to improve the readability). $\mathcal{DB}$ stores all identifiers and secret keys of valid tags. $\mathcal{R}$ communicates with both $\mathcal{T}$ and $\mathcal{DB}$ and provides a link between them. $\mathcal{DB}$ might be considered as apart of the reader. Moreover, $\mathcal{T}$ has a restricted memory and computational capacities and can communicate with $\mathcal{R}$ for the limited distance. We assume that $\mathcal{R}$ is much more talented than the tag. An adversary $\mathcal{A}$ can corrupt a tag and use its internal secrets against the system but she cannot corrupt $\mathcal{R}$. We also assume that the communications between $\mathcal{R}$ and $\mathcal{DB}$ is protected by a secure channel such as Transport Layer Security (TLS) / Secure Sockets Layer (SSL) .

3.1. Definitions of RFID Scheme

An RFID system is defined by the following procedures.

- SetupReader$(1^{\alpha}) \longrightarrow (K_P, K_S)$ is a setup algorithm that generates a public-private key pair (K_P, K_S) for the reader $\mathcal{R}$ where α is the security parameter, then initializes an empty database $\mathcal{DB}$ to store all identifiers and secret keys of all tags. Although K_s is secretly kept in the $\mathcal{DB}$ with the security parameter α , K_p is publicly released.

- SetupTag$(K_p, ID) \longrightarrow (K, S)$ is a probabilistic algorithm which returns a tag secret K and the initial state S of a tag $\mathcal{T}$ with the input identifier ID. When $\mathcal{T}$ is legitimate, the pair $(ID,\ K)$ is to be stored into the database, $\mathcal{DB}$.

- Ident$\longrightarrow$ *Output* is an interaction protocol between a tag $\mathcal{T}$ and the reader $\mathcal{R}$ to complete the protocol transcripts. At the end of the protocol, if $\mathcal{T}$ is legitimate, $\mathcal{R}$ accepts the tag ($\mathcal{R}$ identifies $\mathcal{T}$) and outputs its identifier $Output = ID$, otherwise $\mathcal{R}$ refuses $\mathcal{T}$, if it is not valid and outputs $\perp$.

3.2. Definitions of the Oracles

An adversary $\mathcal{A}$ against an RFID scheme that acts as a honest reader and/or a honest tag to attack the system. We assume that there is only one legitimate reader $\mathcal{R}$ in the RFID system and the both valid reader and tag parties of the system have not prior information about the entity that is interacting with themselves. We also suppose that each experiment always starts with executing

the algorithm SetupReader thus K_p, K_s and 1^α are already generated. We consider that K_p and 1^α are already given to $\mathcal{A}$ but K_s is kept secret because $\mathcal{R}$ cannot be corrupted. Furthermore, we next assume that there are no tags in the system at the beginning of the each experiment and $\mathcal{A}$ is allowed to call $\mathcal{O}^{CreateTag}$ oracle to add new tags to the system.

According to the Vaudenay's model [12], a tag is considered as either a free tag or a drawn. Drawn tags are the set of tags that $\mathcal{A}$ has a visual contact and communicates with them. $\mathcal{A}$ cannot interact with the other free tags. When $\mathcal{A}$ calls $\mathcal{O}^{CreateTag}$ oracle, she generates a new tag whose status is free. The following oracles are used by the adversary $\mathcal{A}$ to interact with the RFID system. First of all, $\mathcal{A}$ setups a new tag of identifier ID.

- $\mathcal{O}^{CreateTag}(ID, b)$: It creates a free tag $\mathcal{T}$ with a unique identifier ID by using SetupTag. $\mathcal{T}$ is legitimate when $b = 1$, otherwise $b = 0$ and $\mathcal{T}$ is not valid. It also inserts (ID, K) into $\mathcal{DB}$. b is implicitly 1 when neglected.

Then, the adversary may change the status of the tag from free to drawn by calling the following oracle.

- $\mathcal{O}^{DrawTag}(dist) \rightarrow (\psi_{\mathcal{T}_1}, b_1, \ldots, \psi_{\mathcal{T}_n}, b_n)$: It randomly selects n free tags among all existing ones with distribution probability of the given *distr*. The oracle assigns a new pseudonym, $\psi_{\mathcal{T}_i}$ for each tag and changes their status to drawn. Hence, the oracle returns array of fresh pseudonyms $(\psi_{\mathcal{T}_1}, \psi_{\mathcal{T}_2}, \ldots, \ldots, \psi_{\mathcal{T}_n})$ of the tags ($\psi_{\mathcal{T}_n}$ is the pseudonym of the n^{th} tag.). The pseudonyms are always changed from session to session so the adversary may interact to drawn tags for only one single session. The relations $(\psi_{\mathcal{T}_i}, ID_i\ )$ are stored in a hidden table *tbl* such that $tbl(\psi_{\mathcal{T}i}) = ID_i$. This oracle also returns a bit array $(b_1, b_2 \ldots, \ldots, b_n)$ where b_i of i^{th} tag whether it is legitimate or not. Furthermore, the oracle may return $\perp$ if the querying tags are already drawn or there is no existing tags.

When the tag is drawn, the adversary is only able to interact to the tag with its pseudonym $\psi_{\mathcal{T}}$. $\psi_{\mathcal{T}}$ is defined as a temporary identifier of a tag and used for pointing the tag anonymously.

- $\mathcal{O}^{Free}(\psi_{\mathcal{T}})$: This oracle changes status of the tag $\mathcal{T}$ that is pointed by the pseudonym $\psi_{\mathcal{T}}$ from drawn to free, then $\mathcal{A}$ is no longer interact with $\mathcal{T}$.

  The secret key of the tag with the pseudonym $\psi_{\mathcal{T}}$ is denoted $key[\psi_{\mathcal{T}}]$. The adversary can corrupt the drawn tags by using the following oracle and obtain the internal values of the tag including its secret key.

- $\mathcal{O}^{Corrupt}(\psi_{\mathcal{T}}) \rightarrow S$: S is the whole memory of the $\psi_{\mathcal{T}}$. $\mathcal{A}$ obtains the $key[\psi_{\mathcal{T}}]$. Eventually, the tag $\mathcal{T}$ with the pseudonym $\psi_{\mathcal{T}}$ is destroyed and $\mathcal{A}$ cannot interact to $\mathcal{T}$ any more.

- $\mathcal{O}^{Launch}() \rightarrow \pi$: It makes the reader $\mathcal{R}$ start a new Ident protocol transcript π.

- $\mathcal{O}^{SendReader}(m,\pi) \rightarrow m'$: This sends the message m to the reader $\mathcal{R}$ in the protocol transcript π and outputs the response m'.
- $\mathcal{O}^{SendTag}(m,\pi) \rightarrow m'$: This sends the message m to $\mathcal{T}$ and outputs the response m'. Also, $\mathcal{A}$ asks for the reader's result of the protocol transcript π. The adversary can use the following oracle to change the status of the tag so she can start to interact with the tag change the status into drawn or she can free the tag and cannot communicate anymore.
- $\mathcal{O}^{Execute}(\psi_{\mathcal{T}}) \rightarrow (\pi, transcript)$: executes a complete protocol between the reader and the tag with pseudonym $\psi_{\mathcal{T}}$. It returns the transcript of the protocol instance that is the list of the all successive messages of the protocol.
- $\mathcal{O}^{Result}(\pi) \rightarrow x$: It returns $x = 1$, when π completes successfully after the IDENT returns $Output \neq \perp$ it means that the tag $\mathcal{T}$ is identified. Otherwise, if $\mathcal{T}$ is not identified and $Output = \perp$,the oracle returns $x = 0$.

Finally, the adversary $\mathcal{A}$ is allowed to obtain the results of the PRNG bit string used in the protocol by a tag $\mathcal{T}$ by querying the following oracle. For simple explanation, π_i denotes i^{th} protocol instance π, s_i is the state of the $PRNG$ of a tag $\mathcal{T}$ for the protocol instance π_i. If $s_i = 0$, $\mathcal{A}$ does not corrupt $\mathcal{T}$ but if $s_i = 1$, she corrupts and captures the $key[\psi_{\mathcal{T}}]$ for the protocol instance π_i. The array of the π_i, s_i values is also denoted by $\theta_\pi \coloneqq \{(s_1,\pi_1),(s_2,\pi_2),\ldots,(s_n,\pi_n)\}$ and θ_π defines the sufficient number of n tuples.

- $\mathcal{O}^{PRNG}(\theta_\pi, \psi_{\mathcal{T}}) \rightarrow (PRNG_1, PRNG_2 \ldots, PRNG_i, \ldots, PRNG_n)$: It outputs the set of the $PRNG$ bit string used on the tag $\mathcal{T}$ with the unique identifier $ID_{\mathcal{T}}$ for the protocol instance π_i and the state s_i. The oracle returns with $\perp$ for any protocol instance π_i, when the PRNG used in this instance cannot be obtained.

$\mathcal{A}$ performs her attack with running an experiment or playing a game by obeying its rules. Firstly, she constructs a RFID system and uses the oracles and gets a result. She wins or looses depending on the following rules.

3.3. Definitions of the Adversary

We define different adversary classes for playing security games. The definition includes Vaudenay's model [12] and our adversary class.

Definition 3.1. ***(Adversary Classes).*** *An adversary $\mathcal{A}$ against to RFID system who has arbitrary number of accesses to the above oracles except $\mathcal{O}^{PRNG}$ oracle.*

- *STRONG $\mathcal{A}$ uses all oracles without any restrictions.*
- *DESTRUCTIVE $\mathcal{A}$ cannot use any oracle against a tag after using $\mathcal{O}^{Corrupt}$ oracle (i.e. the tag has been killed).*

- *FORWARD* $\mathcal{A}$ *can only use* $\mathcal{O}^{Corrupt}$ *oracle after her first call to this oracle.*
- *WEAK* $\mathcal{A}$ *uses all oracles except* $\mathcal{O}^{Corrupt}$ *oracle*

NARROW $\mathcal{A}$ *has no access to* $\mathcal{O}^{Result}$ *oracle.*
RANDOMEYE $\mathcal{A}$ *can access the PRNG oracle* $\mathcal{O}^{PRNG}$*, and reaches the random number(s) used in a tag.*

$$\boxed{WEAK \subseteq FORWARD \subseteq DESTRUCTIVE \subseteq STRONG}$$

3.4. Security Notions

We remind the some security properties of an RFID system such as completeness and soundness.

Definition 3.2. ***(Completeness).*** *An RFID system is complete if the reader* $\mathcal{R}$ *of the system returns the tag identifier* ID *at the end of the protocol (*IDENT*) for a legitimate tag* $\mathcal{T}$ *with very high probability.*

Definition 3.3. ***(Strong Completeness).*** *An RFID system is complete if the reader* $\mathcal{R}$ *of the system returns the tag identifier* ID *at the end of the protocol (*IDENT*) for a legitimate tag* $\mathcal{T}$ *with very high probability although the RFID scheme has been already attacked.*

According to the Vaudenay's model, the security is vital property and should be provided against every attack by the strongest adversary but it is obvious that the security of the scheme is violated when the tag impersonation is occurred if the adversary uses $\mathcal{O}^{Corrupt}$ oracle. Hence, the model permits the adversary to use all oracles except $\mathcal{O}^{Corrupt}$ oracle.

Definition 3.4. ***(Soundness).*** *An RFID system is said sound if an adversary* $\mathcal{A}$ *impersonates a legitimate tag* $\mathcal{T}$ *with negligible probability [14].*

3.5. Privacy

Vaudenay defines privacy notion that is the deducing ability of an adversary to obtain the ID relations of a tag from its protocol instances. He explains *anonymity* and *untraceability* properties under privacy notion that one is about unveiling the ID of tags and the other one is about indistinguishability of any such two tags respectively [12].

In the RFID literature, there are two types of untraceability notions: *forward untraceability* and *backward untraceability.* If an RFID system provides the forward untraceability feature, an adversary $\mathcal{A}$ who compromises a legitimate tag at a time t, cannot trace the future interactions of the tag, $t' > t$. If an RFID system provides the backward untraceability feature, $\mathcal{A}$ also cannot trace past interactions of the tag, $t' < t$. The backward untraceability property is also called as *forward privacy* or *forward secrecy* and this notion is more important than the forward untraceability for real life scenarios.

Vaudenay also considers the privacy of the RFID system based on the adversary classes in Definition 3.1. In his model, he presents a blinded adversary called as blinder $\mathcal{B}$.

Definition 3.5. ***(Blinder, trivial adversary).*** *A blinder $\mathcal{B}$ for an adversary $\mathcal{A}$ is a polynomial-time algorithm that observers the same messages as $\mathcal{A}$ and simulates* Launch, SendReader, SendTag, *and* Result *oracles without having access to the secret keys nor the database of the system. The adversary $\mathcal{A}$ uses the all outputs of the oracles. A blinded adversary $\mathcal{A}^{\mathcal{B}}$ is an adversary who never uses* Launch, SendReader, SendTag, *and* Result *oracles. An adversary $\mathcal{A}$ is said to be trivial if there exists a blinded adversary $\mathcal{A}^{\mathcal{B}}$ such that $\mid Prob[\mathcal{A}\ wins] - Prob[\mathcal{A}^{\mathcal{B}}\ wins] \mid$ is negligible.*

If the success of the simulator and the blind adversary is nearly the same, it means that the blind adversary has higher attack ability at least as the simulator of the system (except using the secret keys). Hence, the authentication and identification of a tag can be considered private. Vaudenay says that an adversary accomplishes his attack (plays a security game) into two phases. In the first phase, she queries the allowed oracles and collects the outputs. In the second following phase, she analyses the obtained results without using any oracle. Between two phases, she also sees the hidden table *tbl* of the $\mathcal{O}^{DrawTag}$ oracle. If she outputs true from her analyzing, then she wins the game.

Definition 3.6. ***(Privacy).*** *An RFID system is P-private if all the adversaries who belong to class P are trivial following Definition 3.5 [12].*

We give the following well-known links between Vaudenay's privacy classes which are obvious.

STRONG	$\Longrightarrow$	DESTRUCTIVE	$\Longrightarrow$	FORWARD	$\Longrightarrow$	WEAK
$\Downarrow$		$\Downarrow$		$\Downarrow$		$\Downarrow$
NARROW-STRONG	$\Longrightarrow$	NARROW-DESTRUCTIVE	$\Longrightarrow$	NARROW-FORWARD	$\Longrightarrow$	NARROW-WEAK

Now we are ready to explain our RANDOMEYE adversary class and the relationship between the other adversary classes. RANDOMEYE adversary class formalizes the weak and/or misuse of the random numbers for real life RFID systems. Tangibly, an adversary $\mathcal{A}$ can query the $\mathcal{O}^{PRNG}$ oracle, she might learn the random numbers used in the authentication protocol. If $\mathcal{A}$ cannot infer the ID of the tag by using the previous information, we consider that the protocol is RANDOMEYE private. It means that the Vaudenay's adversaries classes are not complete and the relationship between them has been change. Therefore, we give the new link for the STRONG class for clear comprehensibility:

RANDOMEYE-STRONG	$\Longrightarrow$	STRONG
$\Downarrow$		$\Downarrow$
NARROW-RANDOMEYE-STRONG	$\Longrightarrow$	NARROW-STRONG

4. Case Study Protocols

In this section, we present a couple of case studies.

4.1. A Case Study Example: Song and Mitchell's Protocol

Firstly, we study on the scheme that was designed by Song and Mitchell (SM) [20] to provide private and secure authentication between low cost RFID tags. Their protocol is depicted below.

In this protocol the reader generates a nonce r_1 and sends it to the tag to start the protocol. The tag receives the nonce and generates a random bit string, r_2 as a temporary secret for the protocol instance. The tag computes $M_1 = r_1 \oplus tid_i$ and $M_2 = f_{tid_i}(r_1 \oplus r_2)$. Then, the tag sends M_1 and M_2 to the reader. The reader evaluates and searches its database by using M_1, M_2 and r_1. If the reader does not find any match, it will stop the session. For the successful match, the reader authenticates the tag and updates the tag information which is $(u_i)_{old}$ and $(tid_i)_{old}$. Then it computes $M_3 = u_i \oplus (r_2 \gg l/2)$ and sends M_3 message to the tag. The tag computes u_i with using M_3 message and checks that $h(u_i) = t_i$. If the checking is matched, the tag authenticates the reader and updates its u_i and t_i values. If the checking is failed, the tag does not update the current values.

We can prove that a RANDOMEYE adversary can trace a tag without corrupting it.

Theorem 1: The SM protocol does not ensure the RANDOMEYE-WEAK privacy.

Proof. An adversary $\boldsymbol{A}$ that performs the following attack.

1. $\boldsymbol{A}$ creates two legitimate tags by using $\mathcal{O}^{CreateTag}(tid_1, 1)$ and $\mathcal{O}^{CreateTag}(tid_2, 1)$ oracles. Then, $\boldsymbol{A}$ draws two tags from the system by calling $\mathcal{O}^{DrawTag}\left(\frac{1}{2}, 2\right)$oracle and obtains two pseudonyms $\mathcal{T}_1$ and $\mathcal{T}_2$. At this point, $\boldsymbol{A}$ does not know tid_1 and tid_2 that are the identifiers of the $\mathcal{T}_1$ and $\mathcal{T}_2$ tags respectively.
2. $\boldsymbol{A}$ calls $\mathcal{O}^{Execute}(\mathcal{T}_1)$ and gets $\theta_\pi = (0, \pi_1)$ for $\mathcal{T}_1$.
3. Then, $\boldsymbol{A}$ requests $\mathcal{O}^{PRNG}[\theta_\pi, \mathcal{T}_1]$ and obtains $(PRNG_1, 1)$ for $\mathcal{T}_1$. For this protocol $PRNG_1$ is equal to the random bit strings r_2 generated by the tag, $\mathcal{T}_1$. $\mathcal{O}^{PRNG}$ oracle performs the following procedures:
   (a) It generates all possible random strings for r_2 with respect to seed of the PRNG used in the tag. Let we call the list as $\boldsymbol{R} = \left[r_2^1, r_2^2, ..., r_2^j, ...r_2^{|K|}\right]$ where $|K|$ is the entropy of the seed.
   (b) It has the list of all the possible $\boldsymbol{X} = \left[tid_1^1, tid_1^2, ..., tid_1^j, ...tid_1^{|K|}\right]$ values with computing $\boldsymbol{X} = M_1 \oplus \boldsymbol{R}$ because M_1 is obtained within the protocol instance.
   (c) Then, it does the exhaustive search to checks the M_2 messages with computing $f_{\boldsymbol{X}}(r_1 \oplus \boldsymbol{R})$. If $M_2 = f_{M_1 \oplus r_2^j}\left(r_1 \oplus r_2^j\right)$, then $\boldsymbol{A}$ obtains the r_2 that is equal to r_2^j.
4. $\boldsymbol{A}$ obtains the tid_1 for $\mathcal{T}_1$ tag with computing $M_1 \oplus r_2$ and updates the internal values of the tag according to the protocol procedure. Therefore, $\boldsymbol{A}$ has the $tid_{1(new)}$ value of the $\mathcal{T}_1$.

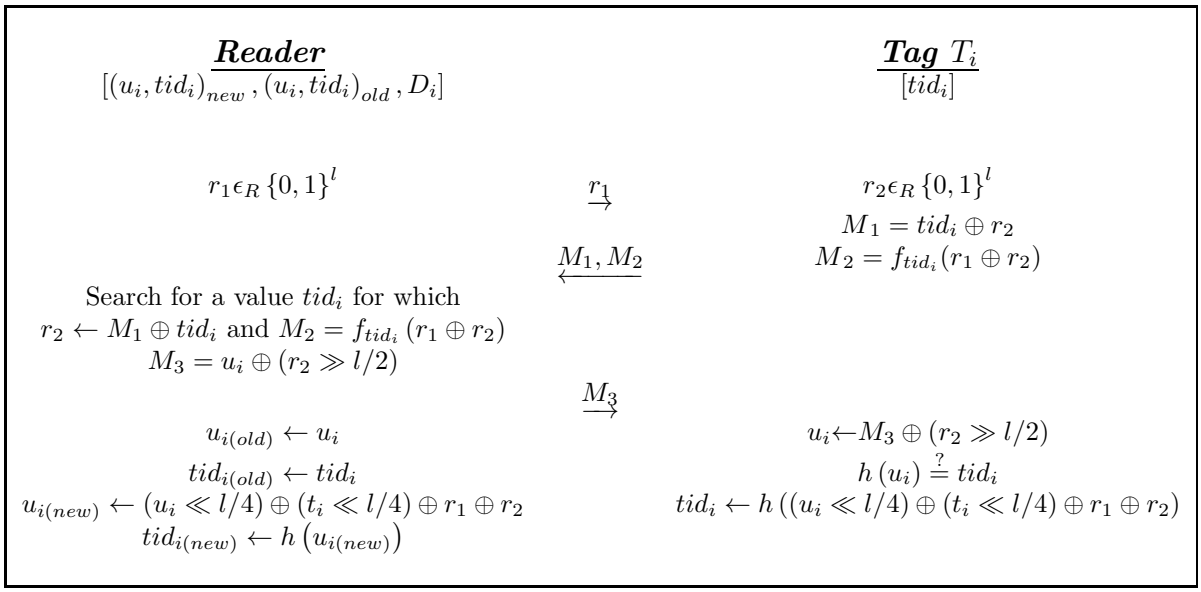

Figure 1: Song and Mitchell's Protocol

5. ***A*** performs the step 2, step 3 and step 4 for the $\mathcal{T}_2$ tag. ***A*** updates the internal values of the tag and gets the $tid_{2(new)}$ value of the $\mathcal{T}_2$.
6. ***A*** frees both tags with request $\mathcal{O}^{Free}(\mathcal{T}_1)$ and $\mathcal{O}^{Free}(\mathcal{T}_2)$, then she reaffects only one of them with using $\mathcal{O}^{DrawTag}\left(\frac{1}{2}, 1\right)$. She obtains a new $\mathcal{T}_3$.
7. ***A*** performs the step 2, step 3 and step 4 for the $\mathcal{T}_3$ tag and obtains the tid_3.
8. Then ***A*** compares tid_3 with $tid_{1(new)}$ and $tid_{2(new)}$.
9. If $tid_3 = tid_{1(new)}$, ***A*** claims that $T_3 = T_1$ else she claims that $T_3 = T_2$.

The success probability of this adversary is definitely 1. Therefore the protocol is not RANDOMEYE-WEAK private.

4.2. Akgun's Scheme

Akgun and Caglayan [47] introduce a new authentication protocol and claim that it is the first protocol provides destructive privacy according to the Vaudenays model with constant identification time. This scheme is a simple challenge/response protocol enhanced with Physically Unclonable Functions (PUFs) in order to achieve higher level of privacy.

This scheme has two phases. In the first phase, the system initializes itself. In this initialization phase, a shared secret S is randomly generated for the back-end server. Two random values, a and b are generated for each tag. Then each tag performs its own PUF $P(.)$ to calculate the $c = S \oplus P(a) \oplus P(b)$. The back-end server stores all values $[ID, a, b, DATA]$ for each tag where $DATA_i$

contains the information about a tag T_i.
In the second phase called authentication phase, the reader generates a random number r_1 and broadcasts at first.
Secondly, a tag T_i which receives the signal of the reader, generates another random number r_2. The tag also computes $M_1 = H(r_1, r_2, a_i)$, $M_1 = H(r_2, r_1, 1) \oplus ID_i$ and $h = H(r_2, 1, 2)$. Then, it uses PUF to calculate $k = P_i(a_i) \oplus r_2$ and deletes the r_2 and $P_i(a_i)$ values from the volatile memory. The tag updates k value with computing $k = k \oplus P_i(b_i) \oplus c_i$ and $P_i(b_i)$ is deleted from the memory too. The tag transmits M_1, M_2 and k back to the reader.
Thirdly, the reader generates a new random number r_3 and computes $r_2' = S \oplus k$, $ID_i' = M_2 \oplus H(r_2', r_1, 1)$. Then, the reader checks that the M_1 message is equal to $H(r_1, r_2', a_i)$ to authenticate the tag T_i. If the equality is checked, then the reader computes $M_3 = H(H(r_2', 1, 2), r_3, b_i)$ and sends r_3, M_3 messages to the tag T_i.
Finally, the tag T_i checks that the M_3 message is equal to $H(h, r_3, b_i)$ to authenticate the reader. If the equality is checked, the tag authenticates the reader too. Thus, mutual authentication is accomplished and the protocol is terminated successfully.

Akgun et al. claims that their protocol scheme provides destructive privacy according to the Vaudenays privacy and security model with constant time identification property. Their protocol does not need key-updating mechanism on both tags and back-end server. The authors uses the common secret S to identify a tag with $\mathcal{O}(1)$ time complexity. They lean the security and privacy of their protocol on the PUFs that have robustness, unclonability, unpredictability and tamper-evident properties [47]. We realize that there is a RNG misuse in their protocol design. We can prove that their protocol is not destructive private and it is not the secure too. A RANDOMEYE adversary can trace the past and future transactions of the tag.

Theorem 2: The Akgun's protocol does not ensure the RANDOMEYE-WEAK privacy.

Proof. An adversary $\boldsymbol{A}$ that performs the following attack.

1. $\boldsymbol{A}$ creates two legitimate tags by using $\mathcal{O}^{CreateTag}(ID_1, 1)$ and $\mathcal{O}^{CreateTag}(ID_2, 1)$ oracles. Then, $\boldsymbol{A}$ draws two tags from the system by calling $\mathcal{O}^{DrawTag}\left(\frac{1}{2}, 2\right)$oracle and obtains two pseudonyms $\mathcal{T}_1$ and $\mathcal{T}_2$. At this point, $\boldsymbol{A}$ does not know ID_1 and ID_2 that are the identifiers of the $\mathcal{T}_1$ and $\mathcal{T}_2$ tags respectively.
2. $\boldsymbol{A}$ calls $\mathcal{O}^{Execute}(\mathcal{T}_1)$ two times and gets $\theta_\pi = \left\{(0, \pi_1^1), (0, \pi_1^2)\right\}$ for $\mathcal{T}_1$.
3. Then, $\boldsymbol{A}$ requests $\mathcal{O}^{PRNG}[\theta_\pi, \mathcal{T}_1]$. $\boldsymbol{A}$ obtains $\left(PRNG_1^1\right)$ and $\left(PRNG_1^2\right)$ respectively for $\mathcal{T}_1$. For this protocol scheme, $PRNG_1^1$ is equal to the random bit strings r_2 generated by the tag, $\mathcal{T}_1$ for the first protocol instance and $PRNG_1^2$ is the secondly generated randıom bit string r_2. $\mathcal{O}^{PRNG}$ oracle performs the following procedures:
   (a) It generates all possible random strings for r_2 with respect to seed of the PRNG used in the tag. Let we call the list as $\boldsymbol{R} = \left[r_2^1, r_2^2, ..., r_2^j, ...r_2^{|K|}\right]$

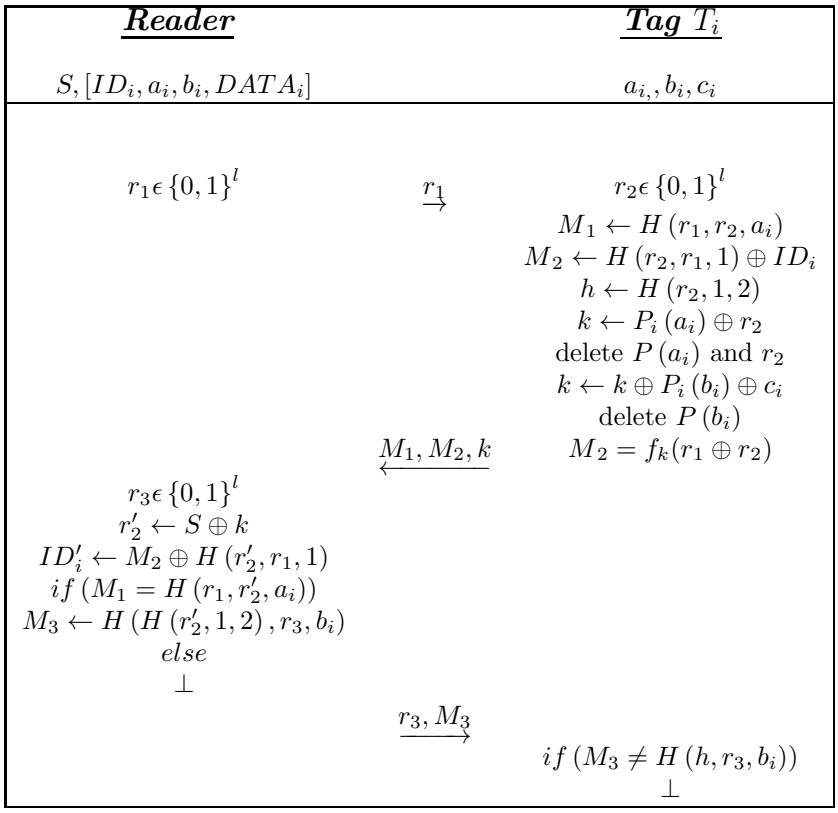

Figure 2: Akgun et al.'s Authentication Protocol

where $|K|$ is the entropy of the seed.

(b) It has the list of all the possible $\boldsymbol{X^1} = \left[ID_1^1, ID_1^2, ..., ID_1^j, ..., ID_1^{|K|}\right]$ values with computing $\boldsymbol{X^1} = M_2 \oplus H(\boldsymbol{R}, r_1, 1)$ because M_2 and r_1 are obtained within the first protocol instance.

(c) It has the second list of all the possible $\boldsymbol{X^2} = \left[ID_1^1, ID_1^2, ..., ID_1^j, ..., ID_1^{|K|}\right]$ values with computing $\boldsymbol{X^2} = M_2 \oplus H(\boldsymbol{R}, r_1, 1)$ because M_2 and r_1 are obtained within the second protocol instance.

(d) Then, it compares the $\boldsymbol{X^1}$ and $\boldsymbol{X^2}$ and defines the identifier of the tag by finding the equal bit string of the each list.

(e) Finally, it obtains the random bit string r_2 by using the corresponding identifier of the tag ID_1 .

4. $\boldsymbol{A}$ obtains the ID_1 for $\mathcal{T}_1$ tag with computing $M_2 \oplus r_2$ with using the one of the protocol instances.
5. $\boldsymbol{A}$ performs the step 2, step 3 and step 4 for the $\mathcal{T}_2$ tag. $\boldsymbol{A}$ obtains the ID_2 for $\mathcal{T}_2$.
6. $\boldsymbol{A}$ frees both tags with request $\mathcal{O}^{Free}(\mathcal{T}_1)$ and $\mathcal{O}^{Free}(\mathcal{T}_2)$, then she reaffects only one of them with using $\mathcal{O}^{DrawTag}\left(\frac{1}{2}, 1\right)$. She obtains a new $\mathcal{T}_3$.
7. $\boldsymbol{A}$ performs the step 2, step 3 and step 4 for the $\mathcal{T}_3$ tag and obtains the ID_3.
8. Then $\boldsymbol{A}$ compares ID_3 with ID_1 and ID_2.
9. If $ID_3 = ID_1$, $\boldsymbol{A}$ claims that $\mathcal{T}_3 = \mathcal{T}_1$ else she claims that $\mathcal{T}_3 = \mathcal{T}_2$.

Therefore, if the adversary $\boldsymbol{A}$ captures the IDs, she can trace the past and future transactions of the tags of the scheme with using the unchanging ID. The scheme does not provide forward and backward untraceability properties.

Theorem 3: The Akgun's protocol does not ensure the RANDOMEYE-DESTRUCTIVE privacy.

Proof. Akgun's protocol does not provide WEAK privacy. Hence, it is not DESTRUCTIVE private 4.2.

Theorem 4: The Akgun's scheme is not secure against RANDOMEYE adversary.

Proof. It is clearly seen that the Akgun's scheme does not provide RANDOM-WEAK privacy and a passive adversary is able to reveal the ID of a tag. Let an adversary $\boldsymbol{A}$ reveals the ID of a tag and consequently has the random bit strings r_2. $\boldsymbol{A}$ also has the k value during the eavesdropping the protocol session where $k = P_i(a_i) \oplus r_2 \oplus P_i(b_i) \oplus c_i$. The shared secret S is generated as $S = P_i(a_i) \oplus P_i(b_i) \oplus c_i$ in the initialization according to the protocol description. Thus, the adversary $\boldsymbol{A}$ obtains the shared secret S by computing $S = k \oplus r_2$. The scheme is not no longer secure after the shared secret S is obtained and the whole system can be broken by the adversary $\boldsymbol{A}$.

5. Conclusion and Future Work

In this paper, we focus on the improper usages of RNGs in privacy-friendly RFID authentication protocols and show that misusing RNGs in a protocol design causes security and privacy weaknesses. To prove our claim, we first have revisited and enhanced an RFID privacy and security model proposed by Vaudenay with modeling a new attack based on misusing of the RNGs. In this context, we extends his model by introducing a new RNG oracle and *RANDOMEYE* adversary class. Then, we apply our model on two recently published lightweight RFID authentication protocols. We show that Song-Mitchell (SM)'s [48] and Akgun-Caglayan's [47] schemes are vulnerable to RNG attack. In our point of view, RNGs should only be utilized to increase the security and privacy level of the protocols instead of becoming a brittle point of the scheme. It is known that a chain is only as strong as its weakest link and we point out that misusage of RNGs might be the weakest link in a protocol design. Moreover, we believe that a new convenient model should be constructed for future analysis instead of Vaudenay's model, although it is more understandable and mature one rather than the earlier models.

References

[1] R. Want, B. N. Schilit, S. Jenson, Enabling the Internet of Things, IEEE Computer 48 (1) (2015) 28–35.

[2] Z. Bilal, Addressing Security and Privacy Issues in Low-Cost RFID Systems, Ph.D. thesis, Royal Holloway, University of London, London, UK, 2015.

[3] F. Armknecht, M. Hamann, V. Mikhalev, Lightweight Authentication Protocols on Ultra-Constrained RFIDs - Myths and Facts, in: N. Saxena, A.-R. Sadeghi (Eds.), Radio Frequency Identification: Security and Privacy Issues, vol. 8651 of *Lecture Notes in Computer Science*, Springer International Publishing, ISBN 978-3-319-13065-1, 1–18, 2014.

[4] A. Juels, Minimalist Cryptography for Low-Cost RFID Tags, in: C. Blundo, S. Cimato (Eds.), International Conference on Security in Communication Networks – SCN 2004, vol. 3352 of *Lecture Notes in Computer Science*, Springer, Amalfi, Italy, 149–164, 2004.

[5] T. Radványi, C. Biró, S. Király, P. Szigetváry, P. Takács, Survey of attacking and defending in the RFID system, Annales Mathematicae et Informaticae 44 (2015) 151–164.

[6] E. M. Garcia, Security Protocols for Low Cost RFID Tags: Analysis and Automated Verification of Proposed Solutions, Tech. Rep., Royal Holloway University of London, Egham, United Kingdom, 2015.

[7] G. Avoine, RFID lounge, http://www.avoine.net/rfid/, [Online; accessed on 15 September], 2016.

[8] Z. Bilal, K. Martin, Q. Saeed, Multiple Attacks on Authentication Protocols for Low-Cost RFID Tags, Applied Mathematics and Information Sciences 9 (2) (2014) 561–569.

[9] S. M. Alavi, K. Baghery, B. Abdolmaleki, Security and Privacy Flaws in a Recent Authentication Protocol for EPC C1 G2 RFID Tags, Advances in Computer Science : an International Journal 3 (5) (2014) 44–52.

[10] G. Avoine, Cryptography in Radio Frequency Identification and Fair Exchange Protocols, Ph.D. thesis, EPFL, Lausanne, Switzerland, 2005.

[11] A. Juels, S. Weis, Defining Strong Privacy for RFID, in: International Conference on Pervasive Computing and Communications – PerCom 2007, IEEE, IEEE Computer Society, New York City, New York, USA, 342–347, 2007.

[12] S. Vaudenay, On Privacy Models for RFID, in: K. Kurosawa (Ed.), Advances in Cryptology ASIACRYPT 2007, vol. 4833 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, ISBN 978-3-540-76899-9, 68–87, URL http://dx.doi.org/10.1007/978-3-540-76900-2_5, 2007.

[13] G. Avoine, Adversary Model for Radio Frequency Identification, Tech. Rep., Swiss Federal Institute of Technology (EPFL), Security and Cryptography Laboratory (LASEC), 2005.

[14] G. Avoine, I. Coisel, T. Martin, Time Measurement Threatens Privacy-Friendly RFID Authentication Protocols, in: S. O. Yalcin (Ed.), Workshop on RFID Security – RFIDSec'10, vol. 6370 of *Lecture Notes in Computer Science*, Springer, Istanbul, Turkey, 138–157, 2010.

[15] J. Ha, S. Moon, J. Zhou, J. Ha, A New Formal Proof Model for RFID Location Privacy, in: Proceding of the 13th European Symposium on Research in Computer Security – ESORICS 2008, vol. 5283 of *Lecture Notes in Computer Science*, Springer, Malaga, Spain, 267–281, 2008.

[16] J. Lai, R. H. Deng, Y. Li, Revisiting Unpredictability-Based RFID Privacy Models, in: Proceedings of the 8th International Conference on Applied Cryptography and Network Security – ACNS 2010, vol. 6123 of *Lecture Notes in Computer Science*, Springer, Beijing, China, 475–492, 2010.

[17] M. Akgün, M. Çağlayan, Extending An RFID Security and Privacy Model by Considering Forward Untraceability, in: Security and Trust Management, Technical University of Denmark, Copenhagen, 239–254, 2011.

[18] S. Kardaş, S. Çelik, M. A. Bingöl, M. S. Kiraz, H. Demirci, A. Levi, k-strong privacy for radio frequency identification authentication protocols based on physically unclonable functions, Wireless Communications and Mobile Computing (2014) 1–17.

[19] J. Hermans, R. Peeters, B. Preneel, Proper RFID privacy: model and protocols, IEEE Transactions on Mobile Computing 13 (12) (2014) 2888–2902.

[20] B. Song, C. J. Mitchell, RFID Authentication Protocol for Low-cost Tags, in: V. D. Gligor, J.-P. Hubaux, R. Poovendran (Eds.), Proceedings of the 1st ACM Conference on Wireless Network Security – WiSec'08, ACM, ACM Press, Alexandria, Virginia, USA, 140–147, 2008.

[21] M. Akgün, M. U. Çağlayan, Towards Scalable Identification in RFID Systems, Wirel. Pers. Commun. 86 (2) (2016) 403–421, ISSN 0929-6212.

[22] H.-Y. Chien, SASI: A New Ultralightweight RFID Authentication Protocol Providing Strong Authentication and Strong Integrity, IEEE Transactions on Dependable and Secure Computing 4 (4) (2007) 337–340.

[23] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, A. Ribagorda, LMAP: A Real Lightweight Mutual Authentication Protocol for Low-Cost RFID Tags, in: Workshop on RFID Security – RFIDSec'06, Ecrypt, Graz, Austria, 12–14, 2006.

[24] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, A. Ribagorda, M2AP: A Minimalist Mutual-Authentication Protocol for Low-Cost RFID Tags, in: J. Ma, H. Jin, L. T. Yang, J. J. P. Tsai (Eds.), International Conference on Ubiquitous Intelligence and Computing – UIC'06, vol. 4159 of *Lecture Notes in Computer Science*, Springer, Wuhan and Three Gorges, China, 912–923, 2006.

[25] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, 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", vol. 4277, Springer, 352–361, 2006.

[26] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, A. Ribagorda, Advances in Ultralightweight Cryptography for Low-cost RFID Tags: Gossamer Protocol, in: K.-I. Chung, K. Sohn, M. Yung (Eds.), Workshop on Information Security Applications – WISA'08, vol. 5379 of *Lecture Notes in Computer Science*, Springer, Jeju Island, Korea, 56–68, 2008.

[27] EPC Global, UHF Air Interface Protocol Standard Generation2/Version2, `http://www.gs1.org/gsmp/kc/epcglobal/uhfc1g2`, [Online; accessed on 6 April 2015], 2014.

[28] P. Peris-Lopez, L. Tong Lee, T. Li, Providing Stronger Authentication at a Low-Cost to RFID Tags Operating Under the EPCglobal Framework, in: C.-Z. Xu, M. Guo (Eds.), Embedded and Ubiquitous Computing - Volume 02 – EUC'08, IEEE, IEEE Computer Society, Shanghai, China, 159–166, 2008.

[29] H.-Y. Chien, C.-H. Chen, Mutual Authentication Protocol for RFID Conforming to EPC Class 1 Generation 2 standards, Computer Standars & Interfaces, Elsevier 29 (2) (2007) 254–259.

[30] G. Avoine, M. A. Bingol, X. Carpent, S. B. Ors Yalcin, Privacy-friendly Authentication in RFID Systems: On Sub-linear Protocols based on Symmetric-key Cryptography, IEEE Transactions on Mobile Computing 99.

[31] G. Avoine, E. Dysli, P. Oechslin, Reducing Time Complexity in RFID Systems, in: B. Preneel, S. Tavares (Eds.), Selected Areas in Cryptography – SAC 2005, vol. 3897 of *Lecture Notes in Computer Science*, Springer, Kingston, Canada, 291–306, 2005.

[32] C. H. Lim, T. Kwon, Strong and Robust RFID Authentication Enabling Perfect Ownership Transfer, in: P. Ning, S. Qing, N. Li (Eds.), International Conference on Information and Communications Security – ICICS'06, vol. 4307 of *Lecture Notes in Computer Science*, Springer, Raleigh, North Carolina, USA, 1–20, 2006.

[33] T. Van Le, M. Burmester, B. de Medeiros, Universally Composable and Forward-secure RFID Authentication and Authenticated Key Exchange, in: F. Bao, S. Miller (Eds.), ACM Symposium on Information, Computer and Communications Security – ASIACCS 2007, ACM, ACM Press, Singapore, Republic of Singapore, 242–252, 2007.

[34] A. J. Menezes, P. C. V. Oorschot, S. A. Vanstone, R. L. Rivest, Handbook of Applied Cryptography, 1997.

[35] W. Schindler, W. Killmann, Evaluation Criteria for True (Physical) Random Number Generators Used in Cryptographic Applications, in: Revised Papers from the 4th International Workshop on Cryptographic Hardware and Embedded Systems, CHES '02, Springer-Verlag, London, UK, UK, ISBN 3-540-00409-2, 431–449, 2003.

[36] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, A. Ribagorda, LAMED - A PRNG for EPC Class-1 Generation-2 RFID specification , Computer Standards and Interfaces 31 (1) (2009) 88 – 97, ISSN 0920-5489.

[37] J. Meli Segu, J. Garcia-Alfaro, J. Herrera-Joancomart, J3Gen: A PRNG for Low-Cost Passive RFID, Sensors 13 (3) (2013) 3816–3830, ISSN 1424-8220, URL `http://www.mdpi.com/1424-8220/13/3/3816`.

[38] A. Peinado, J. Munilla, A. Fúster-Sabater, EPCGen2 Pseudorandom Number Generators: Analysis of J3Gen, IACR Cryptology ePrint Archive 2013 (2013) 825.

[39] J. Garcia-Alfaro, J. Herrera-Joancomart, J. Meli Segu, Remarks on Peinado et al.al's Analysis of J3Gen, Sensors 15 (3) (2015) 6217–6220, ISSN 1424-8220.

[40] J. Meli Segu, J. Garcia-Alfaro, J. Herrera-Joancomart, A Practical Implementation Attack on Weak Pseudorandom Number Generator Designs for EPC Gen2 Tags, Wireless Personal Communications 59 (1) (2011) 27–42, ISSN 0929-6212.

[41] W. Che, H. Deng, W. Tan, J. Wang, A Random Number Generator for Application in RFID Tags, in: P. H. Cole, D. C. Ranasinghe (Eds.), Networked RFID Systems and Lightweight Cryptography, Springer Berlin Heidelberg, ISBN 978-3-540-71640-2, 279–287, URL `http://dx.doi.org/10.1007/978-3-540-71641-9_16`, 2008.

[42] F. Garcia, G. de Koning Gans, R. Muijrers, P. van Rossum, R. Verdult, R. Schreur, B. Jacobs, Dismantling MIFARE Classic, in: S. Jajodia, J. Lopez (Eds.), Computer Security - ESORICS 2008, vol. 5283 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, ISBN 978-3-540-88312-8, 97–114, 2008.

[43] ISO/IEC, Standard 18000RFID Air Interface Standard, `http://www.hightechaid.com/standards/18000.htm`, [Online; accessed on 14 September 2015], 2014.

[44] hashcat, Performance, `http://hashcat.net/oclhashcat/`, [Online; accessed on 30 August], 2015.

[45] P. Bayon, L. Bossuet, A. Aubert, V. Fischer, F. Poucheret, B. Robisson, P. Maurine, Contactless Electromagnetic Active Attack on Ring Oscillator Based True Random Number Generator, in: W. Schindler, S. Huss (Eds.), Constructive Side-Channel Analysis and Secure Design, vol. 7275 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, ISBN 978-3-642-29911-7, 151–166, 2012.

[46] S. Sarma, S. Weis, D. Engels, RFID Systems and Security and Privacy Implications, in: B. Kaliski, c. Kaya ço, C. Paar (Eds.), Cryptographic Hardware and Embedded Systems – CHES 2002, vol. 2523 of *Lecture Notes in Computer Science*, Springer, Redwood Shores, California, USA, 454–469, 2002.

[47] M. Akgün, M. U. Çağlayan, Providing destructive privacy and scalability in RFID systems using PUFs, Ad Hoc Networks 32 (2015) 32 – 42, ISSN 1570-8705.

[48] B. Song, C. J. Mitchell, RFID Authentication Protocol for Low-cost Tags, in: V. D. Gligor, J.-P. Hubaux, R. Poovendran (Eds.), Proceedings of the 1st ACM Conference on Wireless Network Security – WiSec'08, ACM, ACM Press, Alexandria, Virginia, USA, 140–147, 2008.