50 Ways to Break RFID Privacy

Ton Van Deursen

50 ways to break RFID privacy

Ton van Deursen[⋆]

University of Luxembourg
ton.vandeursen@uni.lu

Abstract. We present a taxonomy of attacks on user untraceability in RFID systems. In particular, we consider RFID systems in terms of a layered model comprising a physical layer, a communication layer, and an application layer. We classify the attacks on untraceability according to their layer and discuss their applicability.
Our classification includes two new attacks. We first present an attack on the RFID protocol by Kim et al. targeting the communication-layer. We then show how an attacker could perform an application-layer attack on the public transportation system in Luxembourg.
Finally, we show that even if all of his tags are untraceable a person may not be untraceable. We do this by exhibiting a realistic scenario in which the attacker uses the RFID profile of a person to trace him.

Key words: RFID; privacy; untraceability; attacks; taxonomy;

1 Introduction

Radio frequency identification (RFID) systems consist tags, readers, and a backend. RFID *tags* are small, inexpensive devices that communicate wirelessly with RFID *readers*. Most RFID tags currently in use are passively powered and respond to queries from legitimate, but also rogue RFID readers. They allow to uniquely identify everyday items such as passports [1], electronic transportation tickets, and clothes. A key property of RFID systems is that tags can be scanned without the owner's consent and without the owner even noticing it. Therefore, one must ensure that RFID tags embedded in items carried by a person do not reveal any privacy-sensitive information about that person.

A major privacy threat in current RFID systems is that the RFID system maintainer can monitor and profile the behavior of its users. Consider an RFID system used for public transportation e-ticketing such as the Oyster card[1] or the OV-chipkaart[2]. Every time a person uses public transportation a transaction is registered. By collecting this information over a long period of time, the public transportation companies build large databases of privacy-sensitive information.

[⋆] Ton van Deursen was supported by a grant from the Fonds National de la Recherche (Luxembourg).

[1] http://www.tfl.gov.uk/oyster/

[2] http://www.ov-chipkaart.nl/

In some cases, outsiders to the RFID system may also be interested in monitoring and profiling the users of the RFID system. If a person does not want others to know what items he carries, then the RFID tags attached to these items must not reveal this information to unauthorized RFID readers. For instance, some people may not want to reveal the kind of underwear they are wearing, the amount of money in their wallet, their nationality, or the brand of their watch. Therefore, RFID systems must enforce *anonymity*: the property that items and users cannot be identified [2].

An RFID system that satisfies anonymity does not necessarily prevent an attacker from linking two different actions to the same RFID tag. In this work, we study the privacy notion called *untraceability.* To break anonymity, the attacker's goal is to identify the tag and its user. By contrast, to attack untraceability the attacker's objective is to find out that two (or more) seemingly unrelated interactions were with the same tag.

We define untraceability as follows:

Definition 1 (Untraceability). *An RFID system satisfies* untraceability *if an attacker cannot distinguish, based on protocol messages, whether two actions were performed by the same tag or by two different tags.*

If untraceability is not satisfied, an attacker can attribute different actions to one (possibly unknown) tag. By linking one of these actions to the person that carries the tag the attacker effectively traces that person.

Untraceability of RFID tags is hard to achieve for a number of reasons. Due to their small size and the absence of an active power source, RFID tags are severely restricted in the types of computation they can perform. Also, no physical connection is needed for RFID communication, easing deployment of rogue devices by the adversary. Finally, theoretical results by Damgård and Pedersen show that it is impossible to design an RFID system that satisfies efficiency, security, and untraceability simultaneously [3].

The goal of this paper is to study untraceability of RFID systems from the attacker's perspective. Due to the vast number of different RFID systems, no silver-bullet solution to RFID privacy exists yet. It is, therefore, essential to understand how an attacker can break untraceability before deciding what defenses to deploy. We refer to Juels [4] and Langheinrich [5] for a survey of possible defensive techniques to RFID privacy.

Contributions. Our first contribution is a classification of attacks on the untraceability of RFID systems. We describe a layered communication model for RFID communication (Section 2) consisting of a *physical*, a *communication*, and an *application* layer. We classify existing untraceability attacks according to the corresponding layer they attack. Section 4 describes physical-layer attacks, Section 5 describes communication-layer attacks, and Section 6 describes application-layer attacks.

As a second contribution, we describe new attacks on the communication-layer and the application-layer. Section 5.1 presents a communication-layer attack on the RFID protocol by Kim et al. [6] and Section 6.1 describes how an attacker can recover the date and time of the last 5 travels of a person from his public transportation card.

As a last contribution we show in Section 7 that even if all provide untraceability and an individual tag cannot be traced, a person's RFID profile may still allow an attacker to trace him. Such attacks consider only the particular set of tags carried by a person in order to trace him.

2 RFID communication model

The communication flow in an RFID system is commonly described by a set of protocols. These protocols form a layered structure reminiscent of the OSI reference model for computer networks [7]. To classify attacks on untraceability, we separate the following three layers[3] (see Figure 1):

- The *physical layer* is the lowest layer in the model and provides a link between an RFID reader and a tag. Protocols for modulation, data encoding, and anti-collision are implemented in this layer. The physical layer provides the basic interface for transmission of messages between a reader and a tag.
- The *communication layer* implements various types of protocols to transfer information. Protocols implemented in this layer facilitate tasks such as identification or authentication of a device and updates of cryptographic key material stored on a device.
- The *application layer* implements the actual RFID applications used by the user of the system. Application-layer protocols facilitate fetching and interpretation of data, as well as updating the data on a tag. Examples of such data are account and balance information on a public transportation card and the photo on the tag in an e-passport.

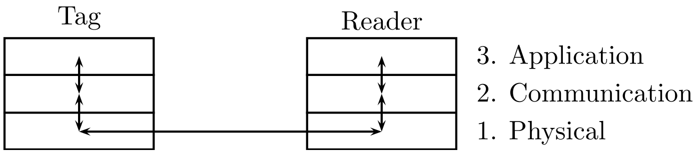

Fig. 1. RFID system layers.

As shown in Sections 4 through 6, each of the layers can leak information that can be used to trace a tag. It is, therefore, important to protect untraceability at every layer of the communication model.

[3] Our model differs slightly from the layered communication model by Avoine and Oechslin [8] since they separate the physical layer into two layers. We additionally introduce an application layer which allows us to reason about high-level attacks.

3 Attacker model

One of the difficulties in designing privacy-preserving RFID systems is that they face powerful attackers. Moreover, the cost of an attack and the knowledge required to perform it are limited. Most equipment necessary to attack RFID systems can be bought for less than $100 and software libraries for most hardware devices are available online. When analyzing RFID systems we assume the attacker has the following capabilities:

- Impersonating readers: A rogue reader can be used for communication with a genuine tag. It implements the same protocol and sends the messages the tag expects to receive.
- Impersonating tags: Similar to impersonating a reader, a rogue tag can be constructed to communicate with a genuine reader.
- Eavesdropping: The attacker captures the transmitted signals using suitable radio frequency equipment [9]. He recovers the transmitted data and listens in on the communication between the reader and the tag. Since the eavesdropping device does not have to power the RFID tag itself, eavesdropping is possible from a larger distance than impersonating a reader.
- Modifying/blocking messages: Although it is hard to carry out in practice, it is possible to relay messages from a legitimate tag to a legitimate reader using a man-in-the-middle device [10]. The man-in-the-middle device can selectively modify transmitted messages, or even block them.

The main difficulty in carrying out attacks is to install the equipment close enough to the legitimate RFID readers and tags. In case of privacy attacks the attacker must carefully install his rogue equipment in a point of interest. Such locations can be entrances to a building, checkout counters of a store, or crowded places. For a discussion on communication distances and eavesdropping distances we refer to Hancke [9].

4 Physical-layer attacks

Physical-layer attacks exploit vulnerabilities that are introduced in the manufacturing process of the RFID tags, the transmission protocols, or the implementation of higher-level protocols. We will first explore a weakness related to the anti-collision identifiers specified by the ISO 14443 [11] standard. We subsequently describe a traceability attack by Danev et al. [12] that abuses the variations in the manufacturing process of RFID tags.

4.1 Static anti-collision identifiers

The greater part of RFID tags currently available implement the physical layer defined by the ISO 14443A standard. Examples of such tags are e-passports, MIFARE tags, and near field communication (NFC) chips. ISO 14443 part 3 describes physical layer protocols for communication with a tag. One of these physical-layer protocols is the anti-collision protocol. The protocol allows the reader to select a particular tag with which it wants to communicate. It prevents communication collisions by ensuring that tags do not respond to the reader simultaneously. It is initiated by the reader after which the tag broadcasts its 32-bit unique identification number (UID).

The anti-collision protocol is not cryptographically protected. Therefore, anybody with an ISO 14443A compliant RFID reader can query a tag for its UID. Almost all currently available ISO 14443A compliant tags have static UIDs. The UIDs cannot be rewritten and never change. Therefore, an attacker can trace a tag (and thus its owner) by repeatedly querying for UIDs. Since static UIDs provide a unique mapping between tags and people, the attacker knows that if the same UID reappears then the same person must be present.

This UID-based traceability attack is very effective in terms of success rate and investment needed. One exception on which the attack outlined above does not work is the e-passport. The e-passport implements randomized UIDs: it is designed to respond with a fresh randomly chosen UID during anti-collision. In terms of implementation costs the attacker needs hardware and software. The hardware needed consists of a computer and an ISO 14443A compliant RFID reader. The latter can be a low-cost off-the-shelf RFID reader currently available for approximately 30 euro[4]. Alternatively, an attacker can use an NFC-enabled phone to carry out the attack. Software to perform the communication between reader and tag can be found online in the form of free software libraries[5].

4.2 Physical fingerprinting

The manufacturing process of RFID tags introduces very small variations in the circuitry of an RFID tag. These variations can be used by an attacker to trace tags. Danev et al. have recently shown that if the radio frequency of the communication is varied, tags of the same brand and type behave differently [12]. Since these differences are stable, an attacker can use them to fingerprint tags and consequently trace tags. Under laboratory conditions and with a small set of tags, the attacks are quite effective. In a set of 50 identical JCOP tags a tag could be correctly recognized in 95% of the cases. The equipment needed by the attacker is relatively expensive and it is hard to perform the attack without being noticed. Therefore, the applicability of the attack is at present quite low.

5 Communication-layer attacks

Communication-layer attacks target the protocols that are used for, among others, identification, authentication, and cryptographic key updates. These protocols are often cryptographic protocols designed to securely authenticate a tag while keeping it untraceable.

[4] `http://www.touchatag.com/`

[5] `http://www.libnfc.org/`

5.1 Unique attributes

In an effort to keep RFID tags cheap, RFID protocols must be computationally as lightweight as possible. Due to the implied absence of strong cryptographic primitives, RFID protocols frequently suffer from algebraic flaws that allow an attacker to perform an *attribute acquisition attack* [13]. In such an attack, the attacker abuses the algebraic properties of the messages exchanged in the protocol to perform a computation that results in a fixed value that is particular to a tag. By repeating this computation at a later stage on different messages and obtaining the same fixed value, the attacker can trace that tag.

We will now restrict ourselves to a subclass of attribute acquisition attacks in which the attack strategy is as follows. Let $f(a, T, i)$ denote the response sent by the tag T upon receipt of its i-th query, where the query equals a. The attacker queries two tags T_1 and T_2 with queries a and a' of his choice and records the responses r and r', where $r = f(a, T_1, i)$ and $r' = f'(a', T_2, j)$. He then performs a computation g that takes the challenge and response as input and satisfies the following conditions:

(a) If T_1 and T_2 are the same tag, then $g(a, r) = g(a', r')$. The attribute $g(a, r)$ is a *unique attribute*;
(b) If T_1 and T_2 are different tags, then $g(a, r) \neq g(a', r')$.

We capture the above intuition in the following definition.

Definition 2 (attribute acquisition attack, adapted from [13]). *Let* Term *be the set of all possible messages of a protocol, let* Tag *be the set of tags in an RFID system, and let $f(a, T, i)$ be the response of tag T in session i upon receipt of query a. We define presence of a unique attribute as follows.*

$$\begin{aligned}&\exists_{T \neq T' \in \mathsf{Tag}}\ \exists_{a, a' \in \mathsf{Term}}\\ &\quad \exists_{i \neq j \in \mathbb{N}}\ \exists_{g : \mathsf{Term}^* \times \mathsf{Tag}\ \mapsto \mathsf{Term}}\\ &\qquad g(a, f(a, T, i)) = g(a', f(a', T, j)) \wedge g(a, f(a, T, i)) \neq g(a', f(a', T', j))\end{aligned}$$

We call $g(a, f(a, T, i))$ a unique attribute*.*

The presence of a unique attribute gives the attacker an efficient way of tracing tags. The attacker merely has to query tags, perform the computation g, and compare the attributes. For a protocol to be untraceable, a necessary condition is that no unique attributes exists. The absence of unique attributes, however, does not guarantee untraceability [13].

An example of an RFID protocol that is vulnerable to an attribute acquisition attack is the protocol proposed by Kim et al. [6] depicted in Figure 2. The protocol is designed to authenticate a tag T to a reader R. Each tag has an identifier ID_T and a key k_T, both known to the reader. The reader initiates the protocol by generating a fresh random value (called a nonce) n. Upon receipt of the query n, the tag generates a nonce s. It then computes the bitwise exclusive-or ($\oplus$) of its identifier ID_T and s as well as the exclusive-or of s and

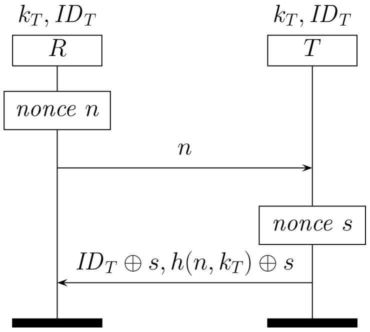

Fig. 2. Privacy protection protocol [6].

the cryptographic hash of n and k_t. The response is then sent to the reader and verified. The exclusive-or function has the following algebraic properties. For any terms a, b, and c and a constant term 0:

$$\begin{array}{ll} a \oplus a = 0 & a \oplus b = b \oplus a \\ a \oplus 0 = a & (a \oplus b) \oplus c = a \oplus (b \oplus c) \end{array} \tag{1}$$

An attribute acquisition attack can be carried out by an attacker that repeatedly queries tags with the same query a. If we let $s_{T,i}$ denote the nonce generated by tag T after the i-th query, then the tag's response to query a is defined by $f(a, T, i) = ID_T \oplus s_{T,i}, h(a, k_T) \oplus s_{T,i}$. A unique attribute can be computed by defining $g(w, (y, z)) = y \oplus z$. To show that $g(a, f(a, T, i))$ is indeed a unique attribute following Definition 2 requires that (a) for two sessions of the same tag, g is the same, and (b) for two sessions of a different tags, g is different. By repetitive application of Equations (1) we obtain:

$$g(a, f(a, T, 0)) = ID_T \oplus s_{T,0} \oplus h(a, k_T) \oplus s_{T,0} = ID_T \oplus h(a, k_T) \tag{2}$$

$$g(a, f(a, T, 1)) = ID_T \oplus s_{T,1} \oplus h(a, k_T) \oplus s_{T,1} = ID_T \oplus h(a, k_T) \tag{3}$$

$$g(a, f(a, T', 1)) = ID_{T'} \oplus s_{T',1} \oplus h(a, k_{T'}) \oplus s_{T',1} = ID_{T'} \oplus h(a, k_{T'}) \tag{4}$$

The term $ID_T \oplus h(a, k_T)$ is a unique attribute for tag T.

5.2 Desynchronization and passport tracing

The following two examples illustrate non-algebraic communication-layer attacks reported in literature.

- One of the first RFID protocols with an untraceability claim was proposed by Henrici and Müller [14]. The protocol relies on a symmetric key that is updated at the end of a successful protocol execution. Avoine showed [15] that the protocol suffered from a number of weaknesses. A particularly interesting attack allowed the attacker to force the reader and tag to perform different key updates, effectively desynchronizing the reader and the tag. As soon as that happens, a genuine reader will no longer be able to successfully complete the protocol and will thus always reject the tag. Assuming that no other tags are desynchronized, carrying out a desynchronization attack on one tag allows the attacker recognize, and thus trace that tag.

- In some RFID systems, an attacker can trace tags by exploiting flaws in the communication-layer and physical-layer simultaneously. Chothia and Smirnov demonstrated [16] that e-passports can be traced by sending a previously observed message to it. It turns out that the e-passport from which the message originated takes significantly longer to respond than a different e-passport would. Therefore, an attacker can trace tags by sending such messages and carefully measuring the time it takes for an e-passport to respond.

6 Application-layer attacks

Application-layer attacks target the application implemented by the RFID system. Therefore, if the RFID system is solely used for identification of items, the application-layer does not implement any protocols. However, RFID tags are becoming more powerful and in some cases the contact interface of a smart card is replaced by a contactless interface using RFID technology. In such cases, the card becomes an RFID tag and care must be taken that the application-layer protocols do not leak any privacy-sensitive information.

6.1 E-go transaction data

The e-go system In 2008, an electronic fare collection system, called e-go, was introduced for public transportation in Luxembourg. E-go is an RFID-based system in which users hold RFID tags and swipe them across RFID readers in buses and on stations. Users can purchase a book of virtual tickets which is loaded on the tag. Upon entering a bus a user swipes his e-go tag and a ticket is removed from it.

Since most RFID readers of the e-go system are deployed in buses the e-go is an *off-line* RFID system [17]. Readers do not maintain a permanent connection with the back-end, but synchronize their data only infrequently. Since readers may have data that is out-of-date and tags may communicate with multiple readers, off-line RFID systems store data on the tags. To store, retrieve, and interpret the data stored on an RFID tag, the RFID system needs to implement application-layer protocols.

The RFID tags used for the e-go system are MIFARE classic 1k tags. These tags have 16 sectors that each contain 64 bytes of data, totaling 1 kilobyte of memory. Sector keys are needed to access the data of each sector. Garcia et al. [18, 19] recently showed that these keys can be easily obtained with off-the-shelf hardware. The data on the tag must therefore be considered to be freely accessible by anybody with physical access to the tag. We thus extend the attacker model from Section 3 to allow attackers to access the data on a tag.

Transaction data Using unprotected tags for public transportation ticketing has obvious security drawbacks. The data can be modified, restored, and even corrupted. Although it is hard to prevent fraud it can be detected and it can be confined by regularly blacklisting abusive tags.

If personal data is stored on an unprotected tag, then the privacy of the user is at stake. On tags in an off-line RFID system such as e-go one expects to find, for instance, the products purchased, the number of unused virtual tickets, and the date and time of the last swipe. A similar fare collection system in The Netherlands stores the date-of-birth of the card-holder and the last 10 transactions on the RFID tag. Researchers have discovered that attackers can recover this data by surreptitiously reading a user's tag [20].

The transaction data provides a history of where the card holder has been on specific dates and times. An attacker could recover this data to profile the users of an RFID system. Such an application-layer attack is more powerful than attacks on the physical layer and communication layer since the attacker does not have to be present when the user swipes his card. He obtains this information from the data stored on the tag.

To understand the transaction data, the attacker must know on which memory location on the tag it is stored and how it is encoded. We will now describe how an attacker can *isolate* and then *decode* the encoded transaction data.

Isolation To recover the address of the transaction data an attacker can use an e-go tag with a book of 10 tickets on it. Upon swiping the tag a ticket is removed and a transaction is written to the tag. A common technique in digital forensics to recover data is to create memory dumps of devices [21]. The attacker can repeatedly swipe the tag and dump the memory to obtain a set of dumps.

A MIFARE 1k tag's memory consists of 16 sectors each of which may contain data. In comparing memory dumps of the tag before and after swipes, only few sectors appear to be updated during a swipe. Five sectors are written to in a cyclic manner and are very similarly structured. It turns out that these five sectors contain the transaction data.

Decoding If we know the location of the date and time information, all that remains is to recover how the date and time are encoded. The encoding can be recovered using the date and approximate time at which the attacker swiped the cards. Table 1(a) gives the raw data and the date of a swipe for a subset of the swipes and Table 1(b) gives similar data for the time of the swipe.

A standard way of encoding date and time is to select a reference date or time and to store the number of days, minutes, seconds, or milliseconds since the reference date or time [22]. The example dates in Table 1(a) are the same if the date of the swipe is the same, but differ by 1 if the date differs by one day. An educated guess suggests that these bits represents the number of days since a particular reference date. Indeed, 01000101001111 in base 2 (4431 in base 10) indicates that the first swipe occurred 4431 days after 01/01/1997: on

Table 1. Sample data for (a) date and (b) time information.

(a) Date

Raw data	Date
01000101001111	18/02/2009
01000101001111	18/02/2009
01000101010000	19/02/2009
01000101010000	19/02/2009
01000101010001	20/02/2009

(b) Time

Raw data	Appr. time
01101101000	14.32
10010001011	19.32
01000000011	08.35
01010011011	11.10
01000000001	08:35

18/02/2009. A similar analysis of the time information in Table 1(b) shows that the time is encoded as the number of minutes elapsed since midnight.

Once the attacker has isolated the date and time and discovered how to decode it, he has a simple procedure of performing an application-layer traceability attack. The attacker needs to have brief physical access to an e-go tag. He can then scan the tag and read its contents with his own hardware. The attacker needs to position his reader reasonably close to the tag. Therefore, crowded areas such as shopping centers or buses, or simply when the user leaves his wallet with e-go tag in his jacket, provide excellent opportunities for an attacker to scan the tag. He then has access to the last 5 transactions stored on the tag. The date and time of these transactions can be recovered by the above decoding. The attacker then knows at what times the owner of the tag has swiped his tag.

6.2 Side-channel information and compositionality

Application-layer attacks are not common in RFID systems, since most RFID systems do not implement application-layer protocols. A more prevalent type of attack is to combine application-layer and communication-layer information to attack privacy. In practice, these attacks are hard to carry out without being noticed since they often require man-in-the-middle hardware to be installed.

- Consider an RFID system that is used for building access. In such a system, the fact that a door opens indicates that the authentication protocol between the tag and the reader was carried out successfully. Such “side-channel” information can sometimes be used by the attacker to attack communication-layer protocols. An example of such an attack is given by Gilbert et al. [23] where the attacker performs a man-in-the-middle attack on a communication protocol and uses the application-layer information to trace a tag.
- An equally complicated attack abuses the fact that an RFID tag implements more than one application, for instance an identification protocol and an ownership transfer protocols. These applications implement different communication-layer protocols P_1 and P_2 each of which could be untraceable in isolation. However, in some situations the messages of protocol P_1 can be combined with messages of P_2 to trace a tag. In [24] a traceability attack that combines messages from a tag-authentication protocol and reader-authentication protocol is described.

7 RFID profiling

In the previous sections, we have described attacks against all layers of the RFID communication model. To maintain the privacy of a user, all these layers must be properly protected. But even if all RFID tags are untraceable, the fact that a person carries a collection of different tags can make him traceable.

Recall that if a tag is untraceable it cannot be distinguished from any other tag of the same type. However, it can be easily distinguished from tags of a different type or brand. An attacker can query a tag with his rogue reader to find out what protocols it runs and hence discover the type of the tag. If everybody carries the same number of tags of the same types the attacker gains no information. However, if people carry different sets of tags an attacker can create a *profile* of them by scanning all their tags and registering their type. The attacker can later recognize a person if he observes the same profile.

The attacks shown in the previous sections abuse design flaws that allow an attacker to trace *one* particular tag. Since only one person carries that tag, the attacker actually traces that person. RFID profiles, however, may be shared among a large set of people. If an attacker observes the same profile twice, he cannot be sure that he observed the same person twice. Therefore, untraceability becomes a probabilistic property. Obviously, if fewer people share the same profile, the probability that two observations of that profile belong to the same person increases. In order to show that the privacy loss due to a person's RFID profile can be significant, we construct and analyze a possible scenario.

7.1 Scenario: United Kingdom

To create a representative data set we use statistical data on inhabitants of the United Kingdom. We study the case where driving licenses, bank cards and store loyalty cards contain RFID tags. We make the following assumptions.

- Each of these RFID tags is untraceable and can, therefore, not be distinguished from other RFID tags of the same type. For instance, a Barclays bank card cannot be distinguished from another Barclays bank card, but it *can* be distinguished from a driver's license or from an HBOS bank card.
- All types of cards are distributed among the population independently at random.
- Unless stated otherwise, the probability that a person carries tag of type A is independent of the probability that he carries a tag of type B for any two types of tags A and B.
- If a person possesses an RFID tag, he will always carry it on him.

Since different types of tags can be distinguished, the fact that a person carries a certain type of tag reduces his privacy. After all, the person can be distinguished from people who do not carry a tag of the same type. A natural way to express privacy loss is by computing the *entropy* of a profile [25]. Entropy expresses the uncertainty of a random variable and we will measure it in bits.

For convenience, we will refer to the entropy of a person instead of the entropy of the random variable associated with the information of a profile.

For instance, there are about 61.6 million inhabitants in the United Kingdom. If we have no identifying information about a random unknown inhabitant then the entropy is $\log_2(61600000) \approx 25.9$ bits. Learning a fact about a person decreases the uncertainty about that person and thus the entropy. If a fact occurs with probability $\Pr[\mathrm{X}]$, the entropy reduction is $-\log_2(\Pr[\mathrm{X}])$ bits.

We will now analyze how much privacy is lost if we know which RFID tags are carried by inhabitants from the United Kingdom.

Driver's licenses According to the Department for Transport, 34.7 million out of 61.6 million inhabitants of the United Kingdom possess a driver's license [26]. Carrying a driver's license thus reduces the entropy by $-\log_2(\Pr[\mathrm{License}]) = -\log_2(34700000/61600000) = 0.82$ bits.

Bank cards There are an estimated 54M checking accounts [27]. The 5 largest banks in terms of market share are Lloyds TSB (19%), RBSG (17%), Barclays (15%), HSBC Group (14%), and HBOS (14%). Nationwide has a market share of 5%. We now assume that exactly one RFID tagged bank card exists for each checking account of these six banks and that the market shares correspond to the number of checking accounts. If inhabitants carry at most one bank card, then carrying a Nationwide card reduces a person's entropy by $-\log_2(\Pr[\mathrm{Nationwide}]) = -\log_2(0.05 \cdot 54000000/61600000) = 4.51$ bits.

Store loyalty cards An estimated 85% of consumers are part of a store loyalty program [28]. For simplicity, we take this to mean that there are $0.85 \cdot 61.6 = 52.4$ million store loyalty cards in circulation distributed among all grocery chains according to their market shares. We assume that only 6 chains have RFID-tagged loyalty cards: Tesco, Asda, Sainsbury's, Morissons, Co-operative, and Netto. Their respective market shares are 30.6%, 16.9%, 15.7%, 11.3%, 9.1%, and 0.8% [29]. In our scenario, inhabitants may go shopping at different grocery stores and may thus carry more than one loyalty card.

The entropy reduction of a person carrying a Co-operative card is thus $-\log_2(0.091 \cdot 0.85) = 3.69$ bits and of a person carrying *no* Tesco card it is $-\log_2(1 - (0.306 \cdot 0.85)) = 0.43$ bits. A person carrying a Co-operative and a Morissons card, but no other store loyalty cards loses 7.95 bits of entropy.

Implications Each of the observations about a person's driver's license, bank card, and loyalty cards reduces the entropy. For instance, a person with a driver's license, a Nationwide card, and Co-operative and Morissons loyalty cards will lose 13.7 bits of entropy. Therefore, only one in every $2^{13.7} \approx 13300$ inhabitants will have the same profile.

The situation becomes worse when people carry "rare" cards. Such cards could be company badges, foreign driver's licenses, or loyalty cards of small stores. In our scenario, a person with no driver's license, a Nationwide bank card, and a Co-operative and Netto loyalty card will lose 17.6 bits of entropy, meaning only one in approximately 200000 will have the same profile.

It is important to note that to carry out an attack that exploits "RFID profiles", no flaw in the design of RFID systems is abused. The attacker only uses the information concerning the types of tags carried by a person to *fingerprint* that person. In our limited scenario, tracing a person is already possible based on profiles of driver's licenses and some bank cards and loyalty cards. Obviously, fingerprinting becomes more effective as more RFID systems are being deployed and people carry more RFID tags on them.

8 Conclusion

The introduction of RFID tags into items we always carry with us has sparked concerns about user privacy. Understanding the attacks against RFID systems is a first step towards defending a person's privacy. We have provided a classification of untraceability attacks according to the RFID system layer they attack. Untraceability can be violated at every layer and must therefore be studied at each layer. We have described two new attacks: one on a communication-layer protocol and one on an application-layer protocol. Finally, we have shown that even if all layers are properly protected, the "RFID profile" of a person may still allow an attacker to trace him.

Acknowledgments. The author thanks Saša Radomirović, Sjouke Mauw, and the anonymous reviewers for valuable comments that helped improve this work.

References

1. Hoepman, J.H., Hubbers, E., Jacobs, B., Oostdijk, M., Wichers Schreur, R.: Crossing borders: Security and privacy issues of the European e-passport. In: IWSEC. Volume 4266 of Lecture Notes in Computer Science., Springer (2006) 152–167
2. Pfitzmann, A., Hansen, M.: A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management (2010) v0.34.
3. Damgård, I., Pedersen, M.Ø.: RFID security: Tradeoffs between security and efficiency. In: CT-RSA. (2008) 318–332
4. Juels, A.: RFID security and privacy: a research survey. IEEE Journal on Selected Areas in Communications **24**(2) (2006) 381–394
5. Langheinrich, M.: A Survey of RFID Privacy Approaches. Personal and Ubiquitous Computing **13**(6) (August 2009) 413–421
6. Kim, I.J., Choi, E.Y., Lee, D.H.: Secure mobile RFID system against privacy and security problems. In: SecPerU 2007. (2007)
7. Zimmermann, H.: OSI reference model — The ISO model of architecture for open systems interconnection. IEEE Transactions on Communications **28**(4) (1980) 425–432
8. Avoine, G., Oechslin, P.: RFID traceability: A multilayer problem. In: Financial Cryptography. Volume 3570 of Lecture Notes in Computer Science., Springer (2005) 125–140

9. Hancke, G.P.: Eavesdropping Attacks on High-Frequency RFID Tokens. In: Workshop on RFID Security – RFIDSec'08. (2008)
10. Hancke, G.P.: Practical attacks on proximity identification systems (short paper). In: IEEE Symposium on Security and Privacy. (2006) 328–333
11. ISO/IEC 14443: Identification cards – Contactless integrated circuit(s) cards – proximity cards (2001)
12. Danev, B., Heydt-Benjamin, T.S., Čapkun, S.: Physical-layer identification of RFID devices. In: USENIX. (2009) 125–136
13. van Deursen, T., Radomirović, S.: Algebraic attacks on RFID protocols. In: WISTP. Volume 5746 of Lecture Notes in Computer Science., Springer (2009) 38–51
14. Henrici, D., Müller, P.: Hash-based enhancement of location privacy for radio-frequency identification devices using varying identifiers. In: PerCom Workshops. (2004) 149–153
15. Avoine, G.: Adversary model for radio frequency identification. Technical Report LASEC-REPORT-2005-001, EPFL (2005)
16. Chothia, T., Smirnov, V.: A Traceability Attack Against e-Passports. In: Financial Cryptography. Lecture Notes in Computer Science, Springer (2010)
17. Garcia, F.D., van Rossum, P.: Modeling privacy for off-line RFID systems. In: CARDIS. (2010) 194–208
18. Garcia, F.D., de Koning Gans, G., Muijrers, R., van Rossum, P., Verdult, R., Schreur, R.W., Jacobs, B.: Dismantling MIFARE classic. In: ESORICS. (2008) 97–114
19. Garcia, F.D., van Rossum, P., Verdult, R., Schreur, R.W.: Wirelessly pickpocketing a MIFARE classic card. In: IEEE Security and Privacy. (2009) 3–15
20. Teepe, W.: In sneltreinvaart je privacy kwijt (in Dutch). Privacy & Informatie (October 2008)
21. Swenson, C., Manes, G., Shenoi, S.: Imaging and analysis of GSM SIM cards. In: IFIP Int. Conf. Digital Forensics. (2005) 205–216
22. Boyd, C., Forster, P.: Time and date issues in forensic computing - A case study. Digital Investigation **1**(1) (2004) 18–23
23. Gilbert, H., Robshaw, M., Sibert, H.: An active attack against HB$^+$ - A provably secure lightweight authentication protocol. Cryptology ePrint Archive, Report 2005/237 (2005)
24. van Deursen, T., Radomirović, S.: EC-RAC: Enriching a capacious RFID attack collection. In: RFIDSec 2010. Volume 6370 of Lecture Notes in Computer Science., Springer (2010) 75–90
25. Eckersley, P.: How unique is your web browser? In: PETS. (2010) 1–18
26. Department of Transport Statistics: Table nts0201: Full car driving licence holders by age and gender: Great Britain, 1975/76 to 2009 (2009) `http://www.dft.gov.uk/pgr/statistics/datatablespublications/nts/`.
27. Office of Fair Trading: Personal current accounts in the UK (2008) `http://www.oft.gov.uk/OFTwork/markets-work/completed/personal/`.
28. Bosworth, M.H.: Loyalty cards: Reward or threat? (2005) `http://consumeraffairs.com/news04/2005/loyalty_cards.html`.
29. TNS Worldpanel: Tesco share turnaround (plus an update on grocery price inflation) (2009) `http://www.tnsglobal.com/news/news-56F59E8A99C8428989E9BE66187D5792.aspx`.