# Insider attacks and privacy of RFID protocols

Ton van Deursen* and Saša Radomirović

University of Luxembourg

**Abstract.** We discuss insider attacks on RFID protocols with a focus on RFID tag privacy and demonstrate such attacks on published RFID protocols. In particular, we show attacks on a challenge-response protocol with IND-CCA1 encryption and on the randomized hashed GPS protocol.
We then show that IND-CCA2 encryption can be used to prevent insider attacks and present a protocol secure against insider attacks. The protocol is based solely on elliptic-curve operations.

## 1 Introduction

Radio frequency identification (RFID) tags are inexpensive devices that communicate wirelessly with RFID readers. Due to their low fabrication cost, their small size, and their ability to uniquely identify an item, RFID tags are found in a myriad of everyday objects. RFID tags can, for instance, be embedded in passports [1], electronic fare tickets [2], library books [3], and clothes [4]. They are often passively powered and respond to any query. Embedding such tags in items we always carry with us leads to privacy concerns. To protect the privacy of the RFID tag bearer while maintaining the tag's functionality, a vast number of cryptographic RFID protocols have been proposed.

Designing a secure and private RFID protocol is difficult for a number of reasons. Tags are computationally limited due to their small size and the absence of an active power source. Implementing full-fledged cryptography often makes tags too expensive or the communication with them too slow. Yet, attackers have a broad range of possibilities to attack an RFID system, from eavesdropping on communications to studying and tinkering with a tag's circuits.

*Insider attacks.* In this paper, we study insider attacks on RFID protocols. Insider attacks are a major source of security breaches of computer systems in general. Some estimates even show that the majority of breaches (70% - 90%) are caused by insiders [5]. One can think of various scenarios for insider attacks. For instance, a malicious merchant may want to cheat one of his customers, a disgruntled employee may want to inflict damage on his employer's assets, or a legitimate user of a system could be compromised and used in an attack against another user. The latter is the case when a computer system is infected with malware or Trojan horses and used to attack another, more important, system.

* Ton van Deursen was supported by the National Research Fund Luxembourg.

Common to all insider attacks is that the adversary abuses the credentials and knowledge of one compromised user to violate a particular security goal of *another* user.

Many cryptographic protocols achieve security in the absence of insider attackers, but fail to achieve their security goals when insider attackers are present. A well-known example is the Needham-Schroeder protocol [6], which was first proven to be secure [7], but later shown to be flawed in the presence of insider attackers [8]. It is therefore not surprising that standard frameworks for security protocol analysis assume that the adversary controls one or more malicious users in the system [9–11].

To perform an insider attack, the adversary needs the key material stored in one legitimate tag. Since RFID tags are often used as hardware tokens, the users of RFID tags usually have no access to the key material. However, one can think of several practical scenarios for the adversary to acquire the key material. For instance, RFID tags are often not sufficiently tamper resistant. If the adversary is a user of the RFID system he can reverse engineer and obtain the key material of one of his own tags. In an entirely different scenario, the adversary could compromise the manufacturer of RFID tags or influence the key generation process for a number of tags.

*Our contribution.* The main goal of this paper is to show the relevance of insider attacks on privacy of RFID protocols and designing a purely elliptic-curve-based protocol that resists insider attacks.

As a starting point, we take the widely used RFID adversary model originally proposed by Vaudenay [12]. We argue that none of the eight adversary classes proposed in the model faithfully represents insider attackers. We characterize the powers of insider attackers by restricting its oracle access. We then show insider attacks on a collection of protocols. More precisely, we show that the randomized hashed GPS protocol [13] does not resist insider attacks. Furthermore, we show that an IND-CCA1 secure encryption scheme is not sufficient to resist insider attacks.

We then design a purely elliptic-curve-based RFID protocol that withstands insider attacks. As a basis, we use a protocol proposed by Vaudenay and the hash-free public key encryption scheme of Cramer and Shoup. The construction of such a protocol is interesting for several reasons. Public key-based protocols aim to maintain privacy against strong attackers. In fact, it can be shown that public-key cryptography is necessary to achieve strong privacy [12]. Asymmetric protocols also enable efficient tag lookup procedures on the reader's side. These are important when there is a large number of tags in the system as well as to defend against timing attacks [14]. Damgård and Pedersen have shown that in a system with symmetric keys, either privacy, security, or efficiency has to be sacrificed [15]. The protocol we present is currently the only protocol based solely on elliptic-curve cryptography that resists insider attacks.

## 2 Preliminaries

### 2.1 RFID privacy

We adopt the RFID privacy model by Hermans et al. [16] which is a refinement of the widely used privacy model by Vaudenay [12]. We give a minimal description of the model and refer the reader to the original paper [16] for full details. The model assumes a reader $R$ and a set of tags $\mathcal{T}$. The adversary $\mathcal{A}$ can interact with the reader and the tags by means of querying oracles.

Tags may either be in the vicinity of the adversary (we call the tag *drawn*) or out of the vicinity of the adversary (*free*). If a tag is drawn, it gets a temporary identity *vtag* through which it can be addressed by the adversary. Initially, all tags are free tags.

The model defines a game-based definition of privacy. The experiment in which the adversary participates is as follows. The experiment chooses a bit $b$ at random. The adversary is then allowed to interact with readers and tags by accessing oracles. At the end of the experiment, the adversary returns a guess for the value of $b$. If the guess is correct, the adversary wins the experiment.

The following eight oracles define the capabilities of the adversary.

- CreateTag $\rightarrow T_i$: creates a new legitimate tag $T_i$.
- Launch $\rightarrow \pi, m$: makes the reader launch a new protocol instance $\pi$ and returns the first message $m$ sent by the reader.
- DrawTag$(T_i, T_j) \rightarrow$ *vtag*: moves the tags $T_i$ and $T_j$ from the set of free tags to the set of drawn tags. A fresh identifier *vtag* is created and the tuple (*vtag*,$T_i$,$T_j$) is stored. The identifier *vtag* can be used to address the tag $T_b$. If $T_i$ or $T_j$ is already drawn, then $\perp$ is returned.
- Free(*vtag*): removes the tuple (*vtag*,$T_i$,$T_j$) and moves tags $T_i$ and $T_j$ to the set of free tags. The tag $T_b$ is rest, that is, its volatile memory is erased.
- SendTag$(m, vtag)_b \rightarrow m'$: looks up the tuple (*vtag*,$T_i$,$T_j$) and sends the message $m$ to $T_i$ (if $b = 0$) or $T_j$ (if $b = 1$). The oracle call returns the tag response $m'$.
- SendReader$(m, \pi) \rightarrow m'$: Sends the message $m$ to the reader's protocol instance $\pi$. If the message $m$ corresponds to the message that the reader expected, he responds with message $m'$. If no active protocol instance $\pi$ exists, $\perp$ is returned.
- Result$(\pi) \rightarrow x$: when the protocol instance with identifier $\pi$ completed successfully, the oracle returns 1, otherwise 0. If no protocol instance $\pi$ exists, $\perp$ is returned.
- Corrupt$(T_i) \rightarrow S$: Returns the state $S$ of the tag. The state contains the current values of the variables mentioned as initial knowledge of the tag. If the tag $T_i$ is drawn, $\perp$ is returned.

**Definition 1 (Privacy).** *A protocol is said to be private if for all polynomial adversaries $\mathcal{A}$, the probability that $\mathcal{A}$ wins the experiment is smaller than $1/2+\epsilon$ for $\epsilon$ negligible in the security parameter.*

The privacy model defines eight adversary classes by restricting access to the CORRUPT and RESULT oracle. The eight classes are obtained by separating four modes of access to the CORRUPT oracle and two modes of access to the RESULT oracle.

The corruption separation is based on the time in the game, at which the attacker may corrupt tags. A *strong* adversary can corrupt tags at any time. The same holds for a *destructive* adversary, with the restriction that he cannot query any other oracles on the corrupted tag. A *forward* adversary can corrupt tags only at the end of the game and a *weak* adversary cannot corrupt tags at all.

A second separation concerns the ability of the attacker to recognize whether a protocol execution between a reader and a tag was successful. In many practical situations this is a reasonable assumption. For instance, in an RFID system for electronic transport tickets, a reader flashing a green light indicates that authentication was successful while a red light indicates it failed. In the model, we call an adversary with access to the RESULT oracle *wide*, while an adversary with no access to the RESULT oracle is called *narrow*.

### 2.2 Elliptic curves and Cramer-Shoup

Let $\mathbb{F}_{2^n}$ be a finite field with $2^n$ elements. Let $\mathcal{E}$ be the group of $\mathbb{F}_{2^n}$-rational points of an ordinary elliptic curve over $\mathbb{F}_{2^n}$. That is, $\mathcal{E}$ denotes the set of points which satisfy the equation $y^2 + xy = x^3 + ax^2 + b$, with $a, b \in \mathbb{F}_{2^n}$ being fixed parameters, together with $\mathcal{O}$, the "point at infinity", which serves as the group's neutral element. We will assume that the group $\mathcal{E}$ contains a subgroup $\mathcal{G}$ of large prime order $p$ and small index in $\mathcal{E}$.

In the following, we recall the elliptic-curve version of the Cramer-Shoup public-key encryption scheme. In section 4.3 we will use and recall the hash-free variant of the scheme.

*Cramer-Shoup public-key encryption scheme.* Let $P_1 \in \mathcal{G}$, $P_1 \neq \mathcal{O}$ and $1 < w, c, d, z < p$ be randomly chosen, system-wide parameters and let $h$ be a collision-resistant hash function. Set $P_2 = wP_1$, $C = cP_1$, $D = dP_1$, $H = zP_1$. The tuple $(P_1, P_2, C, D, H)$ is the RFID reader's public key and $(w, c, d, z)$ its secret key.

To encrypt a message $M \in \mathcal{G}$, we choose a random integer $1 < r < p$ and compute $U_1 = rP_1$, $U_2 = rP_2$, $E = rH + M$, $\alpha = h(U_1, U_2, E)$, and $V = rC + r\alpha D$. The ciphertext is $(U_1, U_2, E, V)$.

To decrypt, correctness of $V$ and $U_2$ needs to be verified first. For this, $\alpha = h(U_1, U_2, E)$ is computed, then $V$ is compared to $cU_1 + \alpha dU_1$ and $U_2$ is compared to $wU_1$. If the terms are equal, then the plaintext is recovered via $M = E - zU_1$.

## 3 Insider attacks

Insider attackers are a class of adversaries whose powers are not accurately represented by any of the eight classes of the privacy model [16, 12]. The four modes of corruption restrict the adversary with respect to *when* he corrupts tags, but not with respect to *which* tags he corrupts. *If* the adversary can corrupt tags he can corrupt *all* tags. We call this type of corruption *full corruption*. In this paper, we are interested in a weaker form of corruption, which we call *selective corruption*. We assume that the attacker can only corrupt his own tags. In particular, the adversary cannot corrupt tags with the purpose of tracing them. Therefore, we only allow corruption of freshly created tags and disallow any queries to corrupted tags. Selective corruption models the case in which an attacker corrupts a tag of his own, with the purpose of tracing some *other* tag in the system.

Although corruption of tags is one way to obtain the keys of a legitimate tag in an RFID system, it is certainly not the only way. Very often, the manufacturer of RFID systems is not the party that implements or deploys the system. In turn, the implementing party is not the same as the maintainer of the system. In practice, before an RFID system is in use, many parties will have possessed the system. All these parties have the possibility to insert tags into the system and later perform insider attacks. For another scenario, consider cell phones with near-field communication (NFC) chips. Nowadays, several cell phones contain NFC chips that can execute RFID protocols. To this end, they contain the key material of RFID tags. If the NFC chip (or possibly the operating system of the cell phone) is compromised, an attacker can obtain the key material of the RFID tags stored on the NFC chip. The key material can then be used in an insider attack. We emphasize that our notion of selective corruption covers both scenarios.

A second restriction we impose on insider attackers is in the way they access the Result oracle. We assume that insider attackers can *only* query the oracle on protocol executions in which no tag was involved. This faithfully models the case where an attacker does not have access to the Result oracle except for the protocol executions which involve only him and an RFID reader.

In the next sections, we demonstrate the relevancy of the class of insider attackers. We show that even under selective corruption and with restricted access to the Result oracle the privacy of several public-key RFID protocols can be broken.

### 3.1 The randomized hashed GPS protocol

The randomized hashed GPS protocol [13] is a privacy-enhancing extension of the GPS identification scheme [17]. Both schemes employ interactive zero-knowledge identification and have been shown to provide strong authentication assuming that the discrete logarithm with short exponents problem is hard. The randomized hashed GPS scheme requires additionally that the decisional Diffie–Hellman problem is hard. The lightweight design of these schemes makes them interesting for low-cost smart cards and RFID tags.

The randomized hashed GPS scheme is based on a group $G$ of prime order, an element $g \in G$, and three integers $A$, $B$, and $S$. The discrete logarithm with short exponents problem and decisional Diffie–Hellman problem are assumed to be hard in $G$ for basis $g$. It further uses a collision and preimage resistant hash function $h$.

Tags have a secret exponent $s \in \mathbb{Z}_S$ and readers a secret exponent $v < 2^k$ (for a security parameter $k$). The public counterpart $g^s$ is known to the reader and $g^v$ is known to the tag. The protocol is initiated with the tag generating two random values $a, b \in \mathbb{Z}_A$. The tag sends $h(g^a, (g^v)^b)$ to the reader upon which it responds with a random challenge $c \in \mathbb{Z}_B$. The tag computes $y = a + b + sc$ and sends it together with $g^a$ and $(g^v)^b$ to the reader. The reader recovers $g^s$ by computing

$$(g^{vy}(g^a)^{-v}((g^v)^b)^{-1})^{1/cv},$$

verifies that the first message equals $h(g^a, (g^v)^b)$, and verifies that $0 \leq y \leq 2A - 2 + (B-1)(S-1)$. The protocol is depicted in Figure 1.

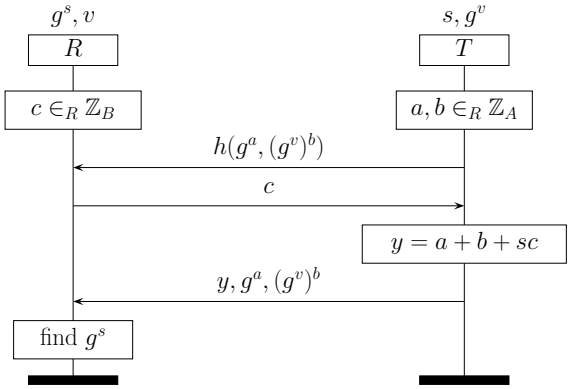


**Fig. 1.** Randomized hashed GPS protocol

The protocol was proven narrow-strong private, wide-forward private, and secure [13, Theorem 5]. We now show that the protocol is vulnerable to insider attacks.

**Theorem 1.** *The randomized hashed GPS protocol (Figure 1) is vulnerable to insider attacks.*

*Proof.* The attack strategy is as follows. The attacker queries two legitimate tags $T$ and $T'$. He combines the tag responses and uses a protocol execution between an insider tag and the reader to verify whether $T = T'$.

We construct an adversary $\mathcal{A}$ that executes the protocol with two legitimate tags $T$ and $T'$. The protocol transcripts for these executions are

$$(h(g^a, (g^v)^b), c, y, g^a, (g^v)^b)$$

and

$$(h(g^{a'}, (g^v)^{b'}), c', y', g^{a'}, (g^v)^{b'}).$$

By the protocol specification, $y$ and $y'$ are defined by $y = (a + b + sc)$ and $y' = (a' + b' + s'c')$. It is the attacker's goal to decide whether $T = T'$ which amounts to deciding whether $s = s'$.

The adversary computes $\alpha$, $\beta$, and $\gamma$ as follows:

$$\begin{aligned} \alpha &= \frac{(g^a)^{c'}}{(g^{a'})^c} = g^{ac'-a'c} \\ \beta &= \frac{((g^v)^b)^{c'}}{(g^v)^{b'})^c} = g^{vbc'-vb'c} \\ \gamma &= c'y - cy' = (ac' - a'c) + (bc' - b'c) + cc'(s - s') \end{aligned} \tag{1}$$

Terms $\alpha$, $\beta$, and $\gamma$ satisfy the following equation if and only if $s = s'$.

$$\alpha \cdot \beta^{1/v} = g^{\gamma} \tag{2}$$

Since the adversary has insider capabilities, he knows the secret $s''$ of one legitimate tag as well as $g^v$ and $g$. To test whether Equation (2) holds, the adversary initiates a protocol execution with a reader. He sends $h(\alpha, \beta)$ upon which the reader challenges with $c''$. The adversary computes $y'' = \gamma + s'' \cdot c''$. If $0 \leq y'' \leq 2A-2+(B-1)(S-1)$, then the adversary sends $y'', \alpha, \beta$ to the reader. In this case, the reader accepts the adversary's insider tag if and only if $s = s'$. Therefore, if the reader accepts the insider tag, we know that $T = T'$, otherwise $T \neq T'$. The protocol flow between the reader and adversary is depicted in Figure 2. For the case that the inequality $0 \leq y'' \leq 2A - 2 + (B - 1)(S - 1)$ is not satisfied, the adversary simply restarts the protocol with the RFID reader. The adversary can avoid this case entirely, if he chooses $c = c' = 1$ and ensures that $\gamma \geq 0$ by swapping, if necessary, $y$ and $y'$ and all other terms from the two protocol transcripts.

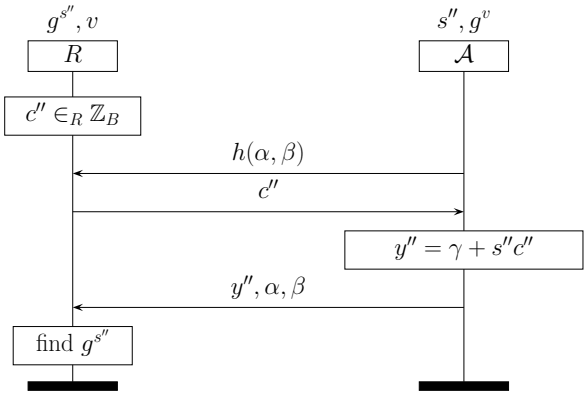


**Fig. 2.** Insider attack on the randomized hashed GPS protocol

Thus, the protocol is vulnerable to insider attacks. □

*Remark 1.* The messages of the EC-RAC protocols [18–20], the randomized Schnorr protocol [21], and the recently proposed hierarchical ECC-based protocol [22] possess homomorphic properties similar to the randomized hashed GPS protocol. It is easy to see that the insider attack shown in the preceding proof can be adapted to all of these protocols and that consequently none of these protocols withstand insider attacks on privacy.

### 3.2 Protocols with IND-CCA1 encryption

In this section, we show that IND-CCA1 encryption is not sufficient to prevent insider attacks. Consider Vaudenay's protocol [12, 16] depicted in Figure 3. The protocol assumes that every pair of reader and tag share a secret key[1] $k$. The reader starts the protocol by sending a random challenge $c$ to the tag. The tag combines the challenge with $k$ and responds with the encryption of $c$ and $k$ under the public key of the reader. The reader decrypts this message with its public key and identifies and authenticates the tag based on the plaintext of the encryption.

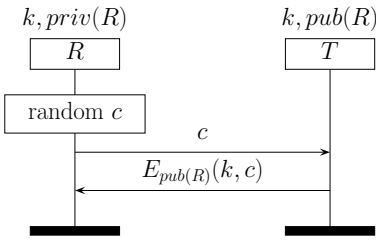


**Fig. 3.** RFID protocol with IND-CCA1 encryption

Hermans et al. [16] show that if the encryption scheme used in the protocol satisfies IND-CCA2, then the protocol is wide-strong private. Since a wide-strong attacker is stronger than insider attackers, it also resists insider attacks.

We show that if a homomorphic IND-CCA1 encryption scheme is used, then the protocol is vulnerable to insider attacks. An encryption scheme is said to be homomorphic if the elements of the plaintext set and the ciphertext set form a group with operators $\otimes$ and $\oplus$, respectively, so that for any encryption key $k$ and for any messages $m_1$ and $m_2$ the encryption function $E_k(\cdot)$ satisfies $E_k(m_1) \oplus E_k(m_2) = E_k(m_1 \otimes m_2)$. Examples of homomorphic encryption schemes are ElGamal [23], DEG [24], and the "lite" version of the Cramer-Shoup scheme [25, Section 5.4]. The latter is an IND-CCA1 scheme which is obtained from the regular Cramer-Shoup scheme (see Section 2) by eliminating the point $D$ and

[1] This key represents the identity ($ID$) and the key ($K$) of the original proposal [12, 16].

the hash function $h$. Thus, to encrypt a message $M \in \mathcal{G}$ with the *lite* Cramer-Shoup scheme, we choose a random $r \in \mathbb{Z}_p$ and compute $U_1 = rP_1$, $U_2 = rP_2$, $E = rH + M$, and $V = rC$. The ciphertext is $(U_1, U_2, E, V)$. Before decrypting, the reader verifies $V = cU_1$ and $U_2 = wU_1$. One can easily see that the scheme is homomorphic if the group operation $+$ is used for each component of the encryption tuple.

**Theorem 2.** *Let $E_{pk}(m)$ denote a homomorphic IND-CCA1 encryption of message $m$ under key $pk$. Then the protocol depicted in Figure 3 does not resist insider attacks.*

*Proof.* By homomorphy of the encryption scheme, we have

$$E_{pub(R)}(k, c) \oplus E_{pub(R)}(k', c') = E_{pub(R)}((k, c) \otimes (k', c')).$$

To attack the scheme, the adversary performs the following insider attack. Suppose tags $T_1$ and $T_2$ share secret keys $k_1$ and $k_2$ with the reader. Clearly, $T_1$ and $T_2$ are the same tag if $k_1 = k_2$. The attacker queries the two tags with the same challenge $c$. The tags return the ciphertexts $E_{pub(R)}(k_1, c)$ and $E_{pub(R)}(k_2, c)$, respectively.

By correctness of the protocol, the two observations concern the same tag if and only if $k_1 = k_2$. The adversary can test this by using his insider tag with key $k_I$ and executing one protocol run with an RFID reader. Say, the reader's challenge is $c''$. The adversary encrypts $E_{pub(R)}(k_I, c'')$ and computes

$$\begin{aligned} E_{pub(R)}(k_1, c) \oplus E_{pub(R)}(k_2, c)^{-1} \oplus E_{pub(R)}(k_I, c'') \\ = E_{pub(R)}((k_1, c) \otimes (k_2, c)^{-1} \otimes (k_I, c'')). \end{aligned}$$

The reader accepts the adversary's response if $k_1 = k_2$ and rejects it otherwise. If the reader accepts the response, the adversary knows that $T_1 = T_2$, otherwise he knows that $T_1 \neq T_2$. Thus, the protocol is not private against insider attacks. □

## 4 A protocol private against insider attacks

We present the first provably wide-strong and authenticating RFID protocol exclusively based on elliptic-curve and scalar operations. Since the insider adversary has a restricted access to oracles that the wide-strong adversary has full access to, it follows that a wide-strong private protocol is also private against insider attacks.

A wide-strong private scheme which only uses elliptic-curve group operations is interesting for two reasons. For one, like any public-key-based scheme, it permits scalable tag identification: As shown by Damgård and Pedersen [15], for symmetric schemes, RFID privacy can only be obtained at the cost of non-scalable tag lookup procedure for the RFID reader. For the other, the implementation of a typical IND-CCA2 public-key cryptosystem on an RFID tag is

quite expensive. To achieve IND-CCA2 security, most cryptosystems rely on three components. An intractable number-theoretic problem, a symmetric block cipher, and a cryptographic hash function. The main cost in such a scheme is incurred by the large number of gates required to implement the number-theoretic operations on one side and an even larger number of gates to implement the cryptographic hash function on the other. Thus, there is interest in reusing the number-theoretic circuits to implement the same functionality provided by a hash function instead of implementing a separate hash function with additional primitives.

Our protocol is an implementation of Vaudenay's public-key protocol which has been proved to be authenticating in [12] and wide-strong private in [16]. The privacy proof requires the protocol to employ an IND-CCA2 public-key encryption scheme.

The encryption scheme we use in the protocol is the hash-free variant of the Cramer-Shoup scheme [25] shown in Section 2. It provides IND-CCA2 security assuming only the decisional Diffie–Hellman assumption.

The RFID protocol requires the RFID tags to encrypt one message for the system's RFID readers, thus one private-key / public-key pair needs to be generated. The RFID readers will store the private key, the RFID tags will be equipped with the public key. The message to be encrypted by the RFID tag consists of the RFID tag's ID and a challenge it receives from the RFID reader. This message needs to be a concatenation of ID and challenge (to avoid algebraic attacks [26]) as well as an element of the group $\mathcal{G}$. Thus we will represent ID and challenge as bit strings and map their concatenation into the group $\mathcal{G}$.

### 4.1 Mapping into the elliptic curve

To map the reader's challenge and the tag's identity into the elliptic curve, we use a simple try-and-increment method [27]. In the following, we identify elements in finite extensions of $\mathbb{F}_2$ with bit strings. Let $k$ be a security parameter. The map $\phi : \mathbb{F}_{2^{n-k}} \to \mathcal{G} \cup \{\mathsf{fail}\}$ is defined as follows. It assigns to $x \in \mathbb{F}_{2^{n-k}}$ an element $(x', y) \in \mathcal{G}$, where the $n-k$ most significant bits of $x' \in \mathbb{F}_{2^n}$ are equal to $x$ and the remaining $k$ bits are such that $(x', y) \in \mathcal{G}$. To find such a pair $(x', y)$ we simply step through all $2^k$ possible bit strings. If no such bit string is found the map returns $\mathsf{fail}$. Since the expected number of try-and-increment steps is 2 [27], the probability of failure is $1/2^{2^k}$. Thus the security parameter $k$ can be fairly small. We refer to [27] for a discussion on how to implement the try-and-increment algorithm securely, that is, resistant to timing attacks.

**Lemma 1.** *If $\mathcal{E}$ has cardinality $2p$, $p$ prime, then the map $\phi$ can be implemented with $2^{k+1}$ computations of the trace function of $\mathbb{F}_{2^n}$ over $\mathbb{F}_2$, and one square root computation over $\mathbb{F}_{2^n}$.*

*Remark 2.* There are several more sophisticated algorithms to map bit strings to points on an elliptic curve than the try-and-increment method we employ above. The most efficient, deterministic maps are Icart's $f_{a,b}$ function [27] and

the SWU map [28–30]. However, special care needs to be taken in order to implement them securely. The $f_{a,b}$ function, for instance, can have up to four elements in the preimage of a point $(x, y)$. For the case of characteristic 2, where the point satisfies the equation $y^2 + xy = x^3 + ax^2 + b$, with $a, b \in \mathbb{F}_{2^n}$ being fixed parameters, these four elements are the solutions of the quartic polynomial $u^4+u^2+xu+(y+a)$ over $\mathbb{F}_{2^n}$. If Icart's function is used in the way our $\phi$ function is used above, an adversary might be able to launch a man-in-the-middle insider attack. The attacker's goal would be that the victim's answer is accepted by the reader if and only if the quartic polynomial contains the adversary's solution as well as the victim's which would identify the victim to the adversary.

### 4.2 The basic protocol

For simplicity, we first demonstrate how to use the regular Cramer-Shoup scheme to implement the protocol. In the next section we will replace the cryptographic hash function by elliptic-curve point operations to obtain a purely elliptic-curve-based protocol.

Let $ID_T$ be a tag $T$'s identity, encoded as a randomly chosen bit string of length $\frac{1}{2}(n-k)$, where $k$ is the security parameter associated with the $\phi$ function. The basic protocol now runs as follows. The reader challenges the tag with a randomly generated bit string $N$ of length $m = \frac{1}{2}(n-k)$. The tag concatenates its identity $ID_T$ with the challenge string $N$ and applies the $\phi$ function to obtain the point $M = \phi(ID_T, N)$ on the elliptic curve. Thus, the tag sends $rP_1, rP_2, rH + \phi(ID_T, N), rC + r\alpha D$ to the reader. The reader accepts the tag if the response verifies correctly. The protocol is depicted in Figure 4 (left).

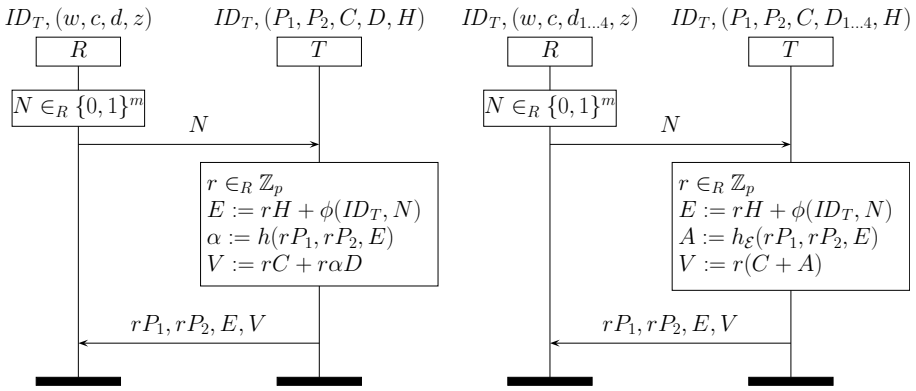


**Fig. 4.** Elliptic-curve-based protocol with a cryptographic hash function (left) and with an elliptic-curve-based hash function (right).

*Correctness, security, and privacy.* Correctness of the scheme follows immediately from correctness of the Cramer-Shoup encryption scheme [25]. Privacy

and security follow Vaudenay [12] and Hermans et al. [16] and the IND-CCA2 security of the Cramer-Shoup encryption scheme.

### 4.3 A purely elliptic-curve-based solution

We now use the hash-free variant of the Cramer-Shoup scheme [25, Section 5.3] to implement a purely elliptic-curve-based protocol. Recall that $\mathcal{E}$ is an elliptic curve over a finite field $\mathbb{F}_{2^n}$ such that it contains a subgroup $\mathcal{G}$ of order $p$, where $p$ is a large prime.

Let $P \in \mathcal{G}$ and $c, d_1, \ldots, d_4, w, z \in \mathbb{Z}_p$ be randomly chosen, system-wide parameters. Set $C = cP$, $D_i = d_i P$ for $1 \leq i \leq 4$, and $H = zP_1$. Then the reader's public key is $(P_1, P_2, C, D_1, \ldots, D_4, H)$ and its secret key is $(c, d_1, \ldots, d_4, w, z)$. Encryption and decryption are as in the regular scheme, shown in Section 2, but the value $\alpha D$ is replaced by $A = h_{\mathcal{E}}(U_1, U_2, E)$, where $h_{\mathcal{E}}$ is a function whose range is a subset of the elliptic curve $\mathcal{E}$. It remains to define the function $h_{\mathcal{E}}$.

*The hash function $h_{\mathcal{E}}$.* The hash function in the encryption scheme needs to hash three points on the elliptic curve onto a single point in a collision-resistant manner. Let $x(Q)$ be the $x$-coordinate of a point $Q$ and $y(Q)$ be its $y$-coordinate. Since $\mathcal{E}$ is a an ordinary elliptic curve over $\mathbb{F}_{2^n}$, a pair $(x(P), y(P))$ is a point on the curve if and only if $(x(P), x(P) + y(P))$ is. Thus for each point on the curve, given its $x$-coordinate, there are only two possible $y$-coordinates. Therefore, only one bit is needed to encode the $y$-coordinate. Furthermore, by Seroussi [31] the $x$-coordinate can be represented within $n-1$ bits, since $\mathcal{G}$ has odd order. We will henceforth refer to this as the $n$-bit encoding of the coordinates. For the point at infinity, we fix an $n$-bit representation.

Define $h_{\mathcal{E}}(X_1, X_2, X_3)$ as follows. For $i = 1, 2, 3$, let $s_i$ be the $n$-bit strings obtained from the $n$-bit encoding of the points $X_1, \ldots, X_3$. We then split the string $s_1, s_2, s_3$ into the four $\lfloor \log_2 p \rfloor$-bit strings $a_1, \ldots, a_4$. Then $h_{\mathcal{E}}(X_1, X_2, X_3) = \sum_{i=1}^{4} a_i D_i$.

Let $ID_T$ be a bit string of length $\frac{1}{2}(n-k)$. The hash-free variant of the protocol is depicted in Figure 4 (right). The protocol can be implemented on an RFID tag with 8 point multiplications, 5 point additions, $2^{k+1}$ computations of the trace function of $\mathbb{F}_{2^n}$ over $\mathbb{F}_2$, and one square root computation over $\mathbb{F}_{2^n}$.

*Correctness, security, and privacy.* Correctness of the scheme follows immediately from correctness of the hash-free Cramer-Shoup encryption scheme. As for the basic version of the protocol, privacy and security follow from Vaudenay [12] and Hermans et al. [16] and the IND-CCA2 security of the hash-free Cramer-Shoup encryption scheme. The latter follows by observing that the $h_{\mathcal{E}}$ function matches the hash function replacement in the hash-free variant of the Cramer-Shoup scheme [25, Section 5.3].

### 4.4 Practicality and different approaches

The protocol presented in the preceding section cannot be considered practical for most applications. There are several aspects to our approach that could be

attempted in a different manner. Our current solution employs four elliptic curve point multiplications to implement the "hash-free" collision resistant function suggested by Cramer and Shoup [25]. The main reason for using a purely elliptic-curve based function is that existing circuits can be reused.

If we allow for hybrid encryption approaches, then a particularly efficient solution would be the OTP-PSEC-3 encryption scheme [32]. This scheme uses two elliptic curve point multiplications and two hash function applications. The scheme has been shown to be IND-CCA2 secure in the random oracle model and based on the elliptic curve gap Diffie–Hellman assumption. One of the two hash functions takes as input a random bitstring, the other takes two bitstrings and two points on the elliptic curve. Using the methods above to produce a purely elliptic-curve based solution, it can be easily seen that the number of point multiplications is at least as large as in our solution.

Finally, a simple way to reduce the complexity of our protocol, albeit at the cost of a rigorous proof of security, is to modify the $\phi$ map as follows. Let $N$ be a scalar and $ID_T$ be a point on the elliptic curve, representing a randomly generated challenge and a tag $T$'s identity, respectively. Then let $\phi(ID_T, N) = NID_T$. The remaining notation and protocol flow are as in Section 4.3 and Figure 4 (on the right). This modification replaces the trace and square root computations of the original $\phi$ map (Lemma 1) by a point multiplication.

## 5 Conclusion

In this paper, we have studied insider attacks on public-key RFID protocols. We have defined insider attackers by restricting the oracle access of a wide-strong adversary in the privacy model by Hermans et al. [16] and Vaudenay [12]. Insider attackers are allowed to corrupt their own tags, but not other tags. Furthermore, insider attackers can only find out whether a protocol execution was successful if it did not involve any legitimate (non-corrupt) tag.

We have shown that insider attacks are a threat to privacy of RFID protocols and we have supported that claim by presenting insider attacks on a number of protocols. Firstly, we have shown an insider attack on the randomized hashed GPS protocol [13]. This attack is more widely applicable and can be performed on other protocols [21, 18–20, 22] as well. Secondly, we have shown that IND-CCA1 cryptosystems are not sufficient to prevent insider attacks. To this end, we used a protocol proposed by Vaudenay which is wide-strong private if an IND-CCA2 secure encryption scheme is used. We have shown an insider attack on the protocol instantiated with an IND-CCA1 secure encryption scheme.

At present, there exists no RFID protocol based solely on elliptic curve cryptography that withstands insider attacks. Motivated by this fact, we have designed the first wide-strong RFID protocol based solely on elliptic curve operations. Since wide-strong attackers are stronger than insider attackers, our protocol resists insider attacks. Although our solution may be too computationally expensive in practice, we stress that it is provably secure, and can therefore serve as a starting point for further research.

Insider attacks are a plausible and important class of attacks, relevant for wide adversaries. In current privacy models, insider attacks are not naturally represented, but can be modeled by assuming a wide-destructive or wide-strong adversary. This is, however, an unreasonable over-approximation of the powers of an insider attacker who cannot corrupt the tags he wants to trace, but only some of his own tags. In the future, we would like to design more efficient protocols that resist insider attacks, but which are not necessarily wide-strong private.

**Acknowledgments.** We thank the anonymous reviewers for valuable comments that helped improve this work. We appreciate in particular a reviewer's reference to the randomized hashed GPS protocol.

## References

1. Juels, A., Molnar, D., Wagner, D.: Security and privacy issues in e-passports. In: Conference on Security and Privacy for Emerging Areas in Communication Networks – SecureComm, IEEE (2005)
2. Sadeghi, A.R., Visconti, I., Wachsmann, C.: User privacy in transport systems based on RFID e-tickets. In: PiLBA. (2008)
3. Molnar, D., Wagner, D.: Privacy and security in library RFID: issues, practices, and architectures. In: ACM Conference on Computer and Communications Security. (2004)
4. Quartararo, P.: Permanent RFID garment tracking system (US Patent 005785181A) (1998)
5. Gollmann, D.: Insider fraud (position paper). In: Security Protocols Workshop. (1998) 213–219
6. Needham, R.M., Schroeder, M.D.: Using encryption for authentication in large networks of computers. Commun. ACM **21**(12) (1978) 993–999
7. Burrows, M., Abadi, M., Needham, R.: A logic of authentication. SIGOPS Oper. Syst. Rev. **23**(5) (1989) 1–13
8. Lowe, G.: Breaking and fixing the Needham-Schroeder public-key protocol using FDR. In: TACAS. (1996) 147–166
9. Lowe, G.: Casper: a compiler for the analysis of security protocols. J. Comput. Secur. **6**(1-2) (1998) 53–84
10. Blanchet, B.: An efficient cryptographic protocol verifier based on Prolog rules. In: 14th IEEE Computer Security Foundations Workshop (CSFW), IEEE Computer Society (2001) 82–96
11. Cremers, C.: Scyther - Semantics and Verification of Security Protocols. Ph.D. dissertation, Eindhoven University of Technology (2006)
12. Vaudenay, S.: On privacy models for RFID. In: Advances in Cryptology - ASIACRYPT 2007. Volume 4833 of Lecture Notes in Computer Science., Kuching, Malaysia, Springer-Verlag (December 2007) 68–87
13. Bringer, J., Chabanne, H., Icart, T.: Efficient zero-knowledge identification schemes which respect privacy. In: ASIACCS. (2009) 195–205
14. Erguler, I., Anarim, E.: Scalability and security conflict for RFID authentication protocols. Cryptology ePrint Archive, Report 2010/018 (2010) `http://eprint.iacr.org/`.

15. Damgård, I., Pedersen, M.Ø.: RFID security: Tradeoffs between security and efficiency. In: CT-RSA. (2008) 318–332
16. Hermans, J., Pashalidis, A., Vercauteren, F., Preneel, B.: A new RFID privacy model. In: ESORICS. (2011) To appear.
17. Girault, M., Poupard, G., Stern, J.: On the fly authentication and signature schemes based on groups of unknown order. J. Cryptology **19**(4) (2006) 463–487
18. Lee, Y.K., Batina, L., Singelée, D., Verbauwhede, I.: Low-cost untraceable authentication protocols for RFID. In: 3rd ACM Conference on Wireless Network Security – WiSec'10. (2010)
19. Lee, Y., Batina, L., Verbauwhede, I.: Untraceable RFID authentication protocols: Revision of EC-RAC. In: IEEE International Conference on RFID – RFID 2009, Orlando, Florida, USA (April 2009) 178–185
20. Lee, Y.K., Batina, L., Singelée, D., Verbauwhede, I.: Wide-weak privacy-preserving RFID authentication protocols. In: The 2nd International Conference on Mobile Lightweight Wireless Systems – Mobilight 2010, Springer-Verlag (2010)
21. Bringer, J., Chabanne, H., Icart, T.: Cryptanalysis of EC-RAC, a RFID identification protocol. In: CANS. (2008) 149–161
22. Batina, L., Seys, S., Singelee, D., Verbauwhede, I.: Hierarchical ECC-based RFID authentication protocol. In: Workshop on RFID Security – RFIDSec'11. (2011) To appear.
23. Gamal, T.E.: A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Transactions on Information Theory **31**(4) (1985) 469–472
24. Damgård, I.: Towards practical public key systems secure against chosen ciphertext attacks. In: CRYPTO. (1991) 445–456
25. Cramer, R., Shoup, V.: A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In: CRYPTO '98: Proceedings of the 18th Annual International Cryptology Conference on Advances in Cryptology, London, UK, Springer-Verlag (1998) 13–25
26. van Deursen, T., Radomirović, S.: Algebraic attacks on RFID protocols. In: Information Security Theory and Practices. Smart Devices, Pervasive Systems, and Ubiquitous Networks (WISTP'09). Volume 5746 of Lecture Notes in Computer Science., Springer (2009) 38–51
27. Icart, T.: How to hash into elliptic curves. In: CRYPTO 2009. Lecture Notes in Computer Science, Springer (2009) 303–316
28. Coron, J.S., Icart, T.: An indifferentiable hash function into elliptic curves. Cryptology ePrint Archive, Report 2009/340 (2009) `http://eprint.iacr.org/`.
29. Shallue, A., van de Woestijne, C.: Construction of rational points on elliptic curves over finite fields. In Hess, F., Pauli, S., Pohst, M.E., eds.: Algorithmic Number Theory, 7th International Symposium, ANTS-VII, Berlin, Germany, July 23-28, 2006, Proceedings. Volume 4076 of Lecture Notes in Computer Science., Springer (2006) 510–524
30. Ulas, M.: Rational points on certain hyperelliptic curves over finite fields. Bull. Pol. Acad. Sci. Math. **55**(2) (2007) 97–104
31. Seroussi, G.: Compact representation of elliptic curve points over $\mathbb{F}_{2^n}$. Technical report, Research Contribution to IEEE P1363 (1998)
32. Okamoto, T., Pointcheval, D.: PSEC-3: Provably secure elliptic curve encryption scheme - V3 (Submission to P1363a). In: IEEE P1363a. (2000)