EAI Endorsed Transactions
on Industrial Networks and Intelligent Systems
Research Article ICST.ORG

Light-weight Key Management Scheme for Active RFID Applications

Mohammad Fal Sadikin[1,*] and Marcel Kyas[1]

[1]AG Computer Systems & Telematics, Freie Universität Berlin, Germany

Abstract

Due to low-cost and its practical solution, the integration of RFID tag to the sensor node called smart RFID has become prominent solution in various fields including industrial applications. Nevertheless, the constrained nature of smart RFID system introduces tremendous security and privacy problem. One of them is the problem in key management system. Indeed, it is not feasible to recall all RFID tags in order to update their security properties (e.g. update their private keys). On the other hand, using common key management solution like standard TLS/SSL is too heavy-weight that can drain and overload the limited resources. Furthermore, most of existing solutions are highly susceptible to various threats reaching from privacy threats, physical attacks to various technics of Man-in-the-Middle attacks. This paper introduces novel key management system, tailored to the limited resources of smart RFID system. It proposes light-weight mutual authentication and identity protection to mitigate the existing threats.

Keywords: RFID and Sensor Node Integration; Key Management System; Security and Privacy; Industrial Applications.

Received on 12 February 2015, accepted on 28 March 2015, published on 17 September 2015

doi: 10.4108/eai.17-9-2015.150286

1. Introduction

The emerging of smart RFID, which is the integration of sensor node to the active RFID system, has been seen as a prominent solution in various fields including industrial applications. Such pervasive computing technology introduces various advantages ranging from low cost, ease of centralized management, practical and comprehensive solution which covers the combination of tracking and sensor applications, to its flexibility to be deployed in large-scale system.

Nevertheless, smart RFID system introduces tremendous security and privacy problems. One of them is the complex problem in large-scale key management system. Indeed, the enforcement of common key management solution like using standard Transport Layer Security (TLS/SSL) requires advanced resources including more memory storage, as well as more communication and computation overhead. Thus, it is infeasible for smart RFID system that associates to limited resources (i.e. limited CPU power, limited memory, limited battery/power, and low bandwidth/data-rate). On the other hand, enforcing manual key management solution by recalling all the RFID tags in order to update the security property (e.g. update the new private key) is infeasible to be applied in large-scale and distributed system. Furthermore, most of existing solutions in key management system for wireless communication are highly susceptible to various security and privacy threats. For instance, an adversary may have chance to perform various techniques of Man-in-the-Middle attacks to compromise the key management system. In this case, an adversary may impersonate as legitimate devices in order to trick the legitimate RFID tag and RFID reader to reveal their sensitive information. In this regards, an adversary may reveal the privacy ranging from the location information, data applications, to the most critical information like security properties (e.g. the private key).

Furthermore, the key management system in large-scale scenario is more complicated since the security properties might be updated frequently. This issue makes the limited resources in the smart RFID tag may suffer manifold request of security update. The following list outlines the conditions that the key management system might require to update the new security properties.

*Corresponding author. E-mail: fal.sadikin@fu-berlin.de

- Life time for the security properties expires. In order to improve the security strength and guarantee the freshness, the security properties should be updated periodically. This method can also mitigate the system from being compromised.
- The increase number of new RFID tag deployed in the existing system might affect the need to update the security properties.
- The compromised tag or malicious tag is detected. In case an adversary is able to compromise one or more RFID tags or RFID readers, all critical security properties in the existing networked system must be changed or updated.

This paper introduces a novel key management system that complements our previous work in RFID-Tate [1]. It is a light-weight key management solution that enables identity protection and mutual authentication using Identity-based Encryption (IBE) method. In particular, it relies on cryptographic Tate (ηT) pairing over super singular elliptic curves, ternary field $F_{3^{509}}$ [2]. Furthermore, in order to prolong the RFID tag lifetime, we propose efficient communication overhead. In this regards, the key management scheme relies on link layer security method, particularly over IEEE 802.15.4 which is commonly used to deliver low-data rate in order to produce efficient processing as well as save the energy. Thus it is affordable to be applied in the constrained nature of smart RFID environment.

1.1. Vulnerabilities

Smart RFID system introduces tremendous problem in security and privacy ranging from the vulnerabilities that arise from the nature of wireless communications, the threats arising from the vulnerable nature of Wireless Sensor Networks (WSN), to the vulnerabilities derived from the use of RFID technic itself. In our previous works [1][3], we already demonstrated that our solution is feasible to mitigate various security and privacy threats in smart RFID system.
In this paper we particularly focus on the threats derived from key management scheme in the constrained nature of smart RFID, which are listed as follow.

- An adversary can perform various technics of Man-in-the-Middle (MITM) attacks in order to hijack the session and intercept the key management system. The adversary may steal the critical security properties including the private key and can use it to perform further malicious activities (e.g. cloning, impersonation, data manipulation, replay attacks, Denial-of-Service (DoS), etc.).
- The smart RFID system is highly susceptible from physical attacks. In his case, the adversary may steal the legitimate RFID tag and subsequently copies all the security properties in order to plant their own RFID tag. The stolen security properties may be used also to perform impersonation or playing with various technics of MITM.
- An adversary can plant their own reader (rogue reader). In this regards, the rogue reader can trick the legitimate RFID tags to reveal their sensitive information. Thus, an adversary can use the revealed information to carry out further malicious activities including impersonation, cloning or revealing the privacy (e.g. the location information).
- An adversary may eavesdrop the key management system in order to elicit the privacy. In this regards, an adversary can find out the sensitive information reaching from the position of the RFID tag to carry out unauthorized tracking, to sensor data applications to perform further malicious activities. An adversary may also collect several parameters to perform resource consumption attack. Such attack aims at draining the limited battery and wasting the limited bandwidth by sending amount of packets to overload the limited resources of the RFID tag.

1.2. Requirements

The following list outlines the important requirements that must be fulfilled in order to achieve the integrity protection in the constrained nature of smart RFID system.

- *Mutual authentication and Authorization*. First of all, all participants in the communication of RFID system must be mutually authenticated before revealing their sensitive information to each other. Thus, it ensures that only authorized RFID tag or reader can be involved in the communication system. This requirement is also important to protect the system from various aforementioned threats.
- *Availability*. In the constrained nature with limited connection and data-rate, the key management solution must ensure that the service is available to the RFID system whenever needed.
- *Privacy*. It ensures the confidentiality of the key management system, which prevents the sensitive information from being eavesdropped or illegally revealed by unauthorized party.
- *Credibility*. It ensures that all messages transported during the key management process are not modified or transited by unauthorized party. Furthermore, this requirement should be able to detect the existence of illegitimated devices (i.e. rogue RFID reader and illegitimate RFID tag).
- *Security strength and resistance*. It ensures that the key management solution is strong enough to prevent various threats ranging from various techniques of brute-force attacks, resource consumption attacks, various techniques of MITM attacks, to the specific threats on RFID communications including cloning, tag emulating, spoofing and impersonation.
- *Communication overhead*. The key management solution must ensure that the size and the number of messages transported during the key management process are affordable for the limited resources (i.e. limited bandwidth or data-rate).

- *Computation overhead*. The key management solution must ensure that the limited resource of RFID tag is feasible to deal with the cryptographic processing.
- *Storage overhead*. Taking into account that smart RFID system also deals with sensor data applications, all security properties that are as well stored in the same place must be roomy enough for the limited memory storage of the smart RFID tag.

1.3. Challenges

The following list describes the challenges in designing efficient key management solution to enforce security and privacy protection in large-scale smart RFID system.

- *Vulnerable nature of wireless communication*. The broadcast nature of wireless channel makes an adversary has good chances to perform active and passive Eavesdroppings including various technics of Man-in-the-Middle attacks.
- *Limited CPU power*. The sensor integration to the RFID tag can give benefit to the increase of processing power. Several standard platforms including Imote2 have diverse options of core frequency (i.e. 104, 208, 312 and 416 MHz). Nevertheless, such CPU options are still not feasible for common solution in standard key management system such as TLS/SSL. Indeed, it introduces high computation overhead that overburdens the limited capabilities of CPU, particularly in large-scale system which the key management activities might be frequently required.
- *Limited battery*. Most of Standard solution like TLS/SSL introduces high communication overhead. This issue causes drainage of the battery energy affecting the lifetime of the RFID tag expires soon.
- *Limited memory storage*. Smart RFID system requires more storage to store various parameter including sensor application data and the security properties. On the other hand, most of standard RFID tag as well as sensor node have very limited memory storage.
- *Low data-rate*. Typically, tiny devices communicate over standard IEEE 802.15.4 which delivers low data-rate. Such standard wireless technology is chosen in order to save the battery energy as well as for efficient processing. Thus, the key management system must sustain this requirement by providing efficient communication that optimally minimizes the number and the length of packets that are transported during the security management process.
- *Large-scale system*. Taking into account the smart RFID tag deployed in large-scale scenario, this issue makes all aforementioned challenges as well as the security management are more complicated.
- *High risk on various security threats*. Most of RFID communication is not mutually authenticated. This issue introduces specific security and privacy problem. In this case, an adversary might perform malicious activities based on various technics of MITM, including cloning attacks, replay attacks, etc. Moreover, an adversary may also intercept the key management system or even revealing the privacy (e.g. tag location) based on the broadcasted MAC Address. Furthermore, the content of RFID tag can easily be read without authorization. This issue introduces high risk on physical attacks. In this regards, an adversary may steal the unsupervised RFID tag and copies the security properties in order to plant their own tag or impersonate as legitimate tag.

2. Related work

Our previous work in IMAKA-Tate [4] defines a novel protocol for mutual authentication and key agreement by using used cryptographic Tate pairing ternary field $F_{3^{509}}$ [2]. In this work, we proved that our protocol is suitable to mitigate various threats, as well as preserve the privacy in the constrained nature of Wireless Indoor Positioning (WIP) applications. In subsequent work [3], we proved that the protocol is also suitable to mitigate various security and privacy threats in the constrained nature of smart RFID system. We improved the performance of IMAKA-Tate by introducing RFID-Tate [1], which is tailored to the limited resources of active RFID tag over IEEE 802.15.4. Nevertheless, the aforementioned works are not aware of the specific challenges in key management system for the constrained nature of smart RFID.

Mulkey et al. [5] proposed an Efficient Protocol for Privacy and Authentication in Wireless Networks. In this work, they use IBE based authentication to improve the performance of existing WPA protocol. This paper proved that the protocol computation performance is significantly better than the computation performance of standard RSA method. However, the complementary use of WPA method is not suitable for smart RFID communication which associates to limited resources. Furthermore, various aspects including key management update and the risk on various physical attacks are not considered in this work. In addition, the client MAC address and several security parameters are sent in clear text, particularly in the first time authentication. This issue may introduce privacy problem, as well as various threats based on revealed identity. Thus, such a protocol solution is not suitable for emerging security and privacy threats in the constrained nature smart RFID system.

Szczechowiak and Collier [6] proposed an identity-based encryption for heterogeneous sensor networks called TinyIBE. They designed a light-weight authentication scheme tailored to the nature of sensor networks that associated to various ranges of node capabilities in terms of memory storage, processing, data-rate and battery supply. However, important aspects in RFID system like mutual authentication and key management solution are not provided in this work.

3. Key Management Scheme

This section describes in detail our proposed scheme for key management solution in large-scale smart RFID system.

3.1. Preliminaries

We apply ηT pairing with 128 bit extension field $F_{3^{509 \times 6}}$. It is noticed as the fastest pairing method over super singular curve [7] with advanced level security strength. It provides cryptographic protection which is about same security level as 3072 bit of RSA method [8][5].

In the key management scheme, we initially assume that the reader and the RFID tag perform mutual authentication to each other like defined in RFID-Tate [1]. In this regards, they communicate over standard IEEE 802.15.4f, which defines standard wireless Physical (PHY) and Media access control (MAC) for active RFID. Furthermore, we assume that each smart RFID tag is complemented with co-processor, as it is integrated to standard platform like Imote2 that has various options of core frequency (i.e. 104, 208, 312 and 416 MHz).

3.2. Setup Phase

Like the setup procedure described in RFID-Tate, the Key Generation Function (KGF) which is handled by the administrator generates all secret parameters to construct the ηT pairing. The generated parameters include a 128 bit integer master secret key s, where $s \in Z_q^*$. Supersingular elliptic curve define over $F_q^*, where\ F_q^* = F_{3^{509}}$. A random point on elliptic curve P as part of public parameter, where $P \in E(F_q)$. Additional random point as another part of public parameter Q, where $Q \in E(F_q)$ and $Q = sP$. Furthermore, the KGF also generates public parameter $g = e(P, P)$. In this context, e is a function that maps $E(F_{3^{509}})\ x\ E(F_{3^{509}}) \rightarrow F_{3^{509 \times 6}}$. In addition, two more parameters are defined as hash functions. The first parameter is $H1$, it is hash function to convert a binary RFID identity to a 128 bit integer, where $H1 : \{0,1\}^* \rightarrow Z_q^*$. The second one is $H2$, this hash function is to convert a parameter on extension filed $F_{3^{509 \times 6}}$ to a 128 bit integer, where $H2 : F_q \rightarrow \{0,1\}^n$. The KGF also calculates the private key for each smart RFID tag $T = \frac{1}{s+t}P$, where s is master secret key and t = $H1$(*smart RFID tag MAC Address*) is a public key of the RFID Tag. The same way to calculate reader private key $R = \frac{1}{s+r}P$, where r is public key of the reader calculated as $r = H1$(*reader MAC Address*).

With the exception of master secret key s, the KGF then manually preloads all parameters to the smart RFID tag and the reader before the network deployment. In overall the KGF preloads (*Private Key (T or R), e, P, Q, g, H1 and H2*) to each smart RFID Tag and Reader's memory.

3.3. Authentication and Key Update Phase

First of all, the RFID tag and the reader must perform mutual authentication in order to build secure connection. The mutual authentication procedure is conducted according to the RFID-Tate standard depicted in figure 1. The following list outlines the RFID-Tate authentication [1].

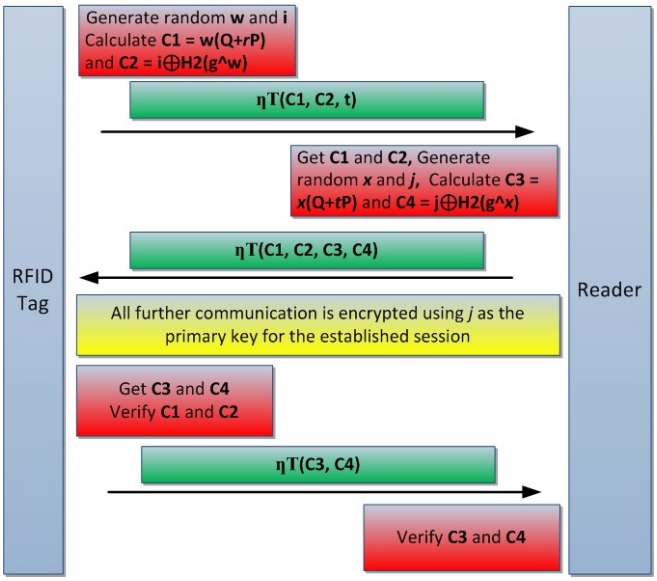

Figure 1. RFID-Tate Scheme, reproduced from [1].

1. We presume that the RFID tag initially sleeps and wakes up after receiving beacon frame broadcasted by the RFID reader. Hereafter, the tag calculates the reader public key as $r = H1(reader\ MAC\ address)$. Subsequently, the tag randomly generates two 128 bit integer i and w, where i is temporary session key.
2. The tag then generates two ciphertexts $C1 = w(Q + rP)$ and $C2 = i \oplus H2(g^w)$. The tag subsequently requests to join in the RFID system by sending the two ciphertexts to the reader. The tag also includes its MAC address $t = H1(tag\ MAC\ address)$ in the encrypted payload, in order to protect its identity from being revealed by unauthorized party. In this case, all contents in the message including the session key i and the tag MAC address are encrypted using the reader public key. Thus, only the reader can decrypt the message.
3. The reader receives and decrypts the messages using its private key R. The reader can recover the session key i by calculating $i = H2(e(R, C1)) \oplus C2$. To achieve efficient communication, the reader tentatively saves the key i and the value of $C1$ for further steps. Each message created by the tag in the three-way handshake will use the initial session key i and the value of $C1$ will be used to calculate primary session key.

The temporary session key is shared based on the pairing function calculated as follows.

$$i = H2(g^w) \oplus C2 \tag{1}$$

since

$$\begin{aligned} e(R, C1) &= e\left(\frac{1}{s+r}P, w(Q + rP)\right) \\ &= e(P, Q + rP)^{\frac{w}{s+r}} \\ &= e(P, (s + r)P)^{\frac{w}{s+r}} \\ &= e(P, P)^w = g^w \end{aligned} \tag{2}$$

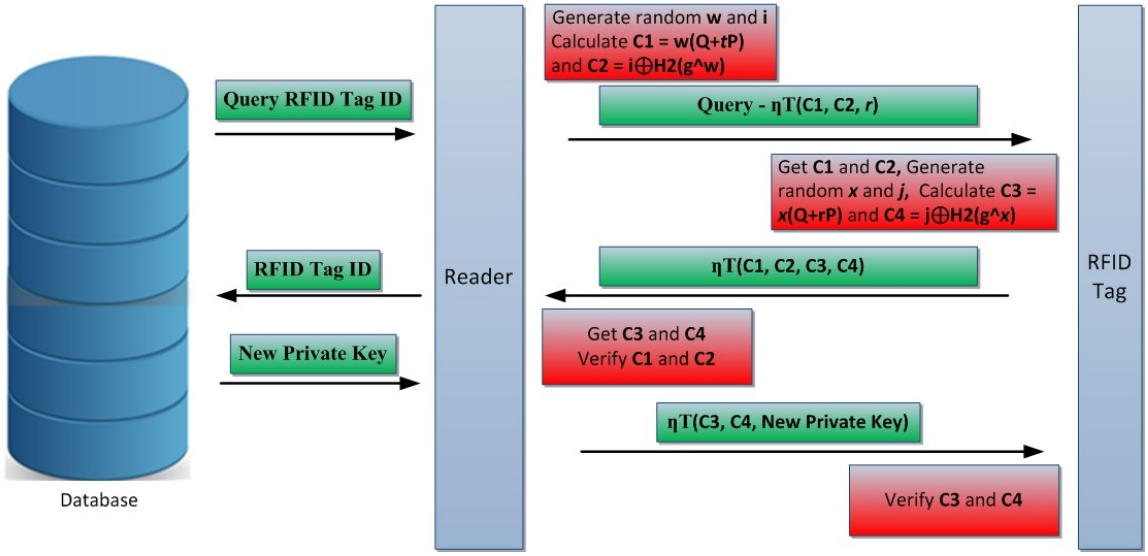

Figure 2. Three-way handshake of the key management scheme.

4. In the second message of the three-way handshake, the reader generate x and j as two random 128 bit integers, where j is primary session key that is used to encrypt all further communication in the established session. The reader afterward generates and send two ciphertexts $C3 = x(Q + tP)$ and $C4 = j \oplus H2(g^x)$. The reader also includes the values of $C1$ and $C2$ in the encrypted message to be further verified by the tag.
5. The tag then receives and decrypts the message which contains temporary session key j using its private key T. It is conducted by calculating $j = H2(e(T, C3)) \oplus C4$. The tag further verifies the value of $C1$ and $C2$. The further step is then continued only if the two values are same as the two values of $C1$ and $C2$ generated by the tag on the first message. Otherwise, the tag aborts the authentication.
6. The tag then sends back the value of $C3$ and $C4$ to be verified by the reader in the encrypted-payload. The further communication is continued if the received values are equal as the values generated by the reader on the second message. Otherwise, the reader sends failure notification to abort the connection. Up to this step, both parties have mutually authenticated to each other.

After the reader and the RFID tag have connected to each other, the database is now ready to carry out key update procedure. In this scenario, the reader acts as a pass-through device, while the RFID tag and the database act as client and server. Figure 2 illustrates the mutual authentication and key update by performing encrypted three-way handshake negotiation. The following list describes the three-way handshake procedure.

1. Initially, the database sends query message to the RFID tag through the reader, in order to inquiry the tag ID. In this case, The reader firstly calculates the tag public key as $t = H1(tag\ MAC\ address)$. Subsequently, the reader randomly generates two 128 bit integer i and w, where i is temporary session key.
2. The reader then generates two ciphertexts $C1 = w(Q + tP)$ and $C2 = i \oplus H2(g^w)$. The reader subsequently passes on the query message that is included in the two ciphertexts to the RFID tag. The reader also includes its MAC address $r = H1(reader\ MAC\ address)$ in the encrypted payload. This mechanism is conducted in order to protect its identity from being revealed by unauthorized party. In this case, all contents in the message including the query message, the session key i and the reader MAC address are encrypted using the RFID tag public key. Thus, only the corresponded RFID tag can decrypt the message.
3. The tag receives and decrypts the messages using its private key T. The tag is able to elicit all contents of the message including the session key i by calculating $i = H2(e(T, C1)) \oplus C2$.

The temporary session key i is shared based on the pairing function calculated as follows.

$$i = H2(g^w) \oplus C2 \tag{3}$$

since

$$\begin{aligned} e(T, C1) &= e\left(\frac{1}{s+t} P, w(Q + tP)\right) \\ &= e(P, Q + tP)^{\frac{w}{s+t}} \\ &= e(P, (s + t)P)^{\frac{w}{s+t}} \\ &= e(P, P)^w = g^w \end{aligned} \tag{4}$$

4. In the second message of the key management process, the tag generate x and j as two random 128 bit integers, where j is primary session key that is used in the rest of key management process including transporting the new private key for the RFID tag. The tag afterward generates and send two ciphertexts $C3 = x(Q + rP)$ and $C4 = j \oplus H2(g^x)$. The tag also attaches the chippertexts of $C1$ and $C2$ in the encrypted message to be verified by the reader.
5. The reader then proceeds the key management process by decrypting the message which contains

the session key *j* using its private key *R*. It is carried out by computing $j = H2(e(R, C3)) \oplus C4$. The reader further verifies the value of *C1* and *C2*. The reader then pass on the tag ID to the database only if the two values are same as the two values of *C1* and *C2* generated by the reader on the first message. Otherwise, the reader discards the query.

6. The database then sends the new private key of the corresponding RFID tag through the reader. The reader then forwards the new private key to the corresponding tag and includes the value of *C3* and *C4* to be verified by the tag in the encrypted-payload. The tag accept the new private key if the received values of *C3* and *C4* are equal as the values generated by the tag on the second message. Otherwise, the reader sends failure notification to discard the key management process.

Up to this step, the RFID tag can use its new private key to perform further mutual authentication with the reader as well as securing the communication according to the RFID-Tate scheme.

4. Computation Overhead Analysis

Efficient computation overhead is a critical requirement for RFID system that associates with limited resources. In order to ensure that our key management solution is affordable for the smart RFID system, we conducted a benchmark test that iteratively calculate the time performance of IBE 128 bit ηT pairing over $F_{3^{509 x 6}}$. This benchmark test is written in C++ adopted from [2]. Furthermore, we also compared our method by performing another benchmark test, which is 3072 bit modulus RSA method with standard OpenSSL library adopted [5]. We executed the benchmarks in our platform under Windows 7 with 64-bit Intel 2 Cores at 1.8 GHz.

On each key management process, each RFID tag computes several cryptographic parameters in order to perform mutual authentication. The parameters are two Multiplication over $F_{3^{509 x 6}}$ (Multipl. $F_{3^{509 x 6}}$), one Exponentiation over $F_{3^{509 x 6}}$ (Exp. $F_{3^{509 x 6}}$) and one ηT Pairing. Table 1 summarizes the computation performance of RFID-Tate at 2 Cores at 1.8 GHz. In addition, Table 2 summarizes the computation performance of RSA 3072 bit modulus.

According to the two benchmark tests in Table I and Table II, RFID-Tate mutual authentication performance is much more efficient than the RSA method. It is to be noted that the benchmark test shown on the RSA method was only to assess the computation overhead for sharing the session key. It depends on the protocol and architecture, the mutual authentication on RSA benchmark test would burden higher computation effort. Furthermore, common RSA method such as standard TLS/SSL relies on upper layer communication as well as requires all time connection to the third-party server called PKI system. This issue requires more expenditure in term of computation and communication overhead. Thus, compared to the common solution like standard TLS/SSL that uses RSA method, our key management solution is obviously more feasible for the constrained nature RFID system.

Moreover, in order to emulate the smart RFID system, we forced the CPU to run in a single core. In addition, we scaled down clock frequency according to the three options of Imote2 platform (i.e. 104 MHz, 208 MHz and 416 MHz). Table 3, 4 and 5 summarize the 1000 iterations of computation performance calculated by each RFID tag at the three options of clock frequencies. According to the three tables, the computation performance is feasible for the smart RFID tag, even it is affordable for the lowest CPU option at 104 MHz which computes the mutual authentication only in 0.057 sec. Moreover, the average performance is more than two times faster when the benchmark test was executed in the advanced clock frequency at 416 MHz. Indeed, the key management process was accomplished only in 0.024 sec.

TABLE I. Estimation of RFID-Tate computation

Phase	Main Operations	Time Estimation
Mutual Authentication & Sharing the new Private Key	1 ηT Pairing	6.426 *ms*
	1 Exp. $F_{3^{509 x 6}}$	0.169 *ms*
	2 Multipl. $F_{3^{509 x 6}}$	0.004 *ms*
	Total	**6.639 *ms***

TABLE II. 3072 bit modulus RSA Computation

Operation	Iterations	Average Time (sec)
Key Generation	50	2681 *ms*
Encryption	5000	0.229 *ms*
Decryption	500	150 *ms*

Table III. Estimation of RFID-Tate at 416 MHz

Phase	Main Operations	Time Estimation
Mutual Authentication & Sharing the new Private Key	1 ηT Pairing	*22.79 ms*
	1 Exp. $F_{3^{509 x 6}}$	*0.577 ms*
	2 Multipl. $F_{3^{509 x 6}}$	*0.181 ms*
	Total	***23.54 ms***

Table IV. Estimation of RFID-Tate at 208 MHz

Phase	Main Operations	Time Estimation
Mutual Authentication & Sharing the new Private Key	1 ηT Pairing	*34.34 ms*
	1 Exp. $F_{3^{509 x 6}}$	*1.144 ms*
	2 Multipl. $F_{3^{509 x 6}}$	*0.451 ms*
	Total	***35.93 ms***

Table V. Estimation of RFID-Tate at 104 MHz

Phase	Main Operations	Time Estimation
Mutual Authentication & Sharing the new Private Key	1 ηT Pairing	*54.54 ms*
	1 Exp. $F_{3^{509 x 6}}$	*2.129 ms*
	2 Multipl. $F_{3^{509 x 6}}$	*0.653 ms*
	Total	***57.32 ms***

5. Security Analysis

This section presents the security proof of our key management solution that prevents an authorized party to elicit the sensitive information as well as mitigating various threats arises in the constrained nature of smart RFID. Moreover, we also demonstrate that our solution can mitigate the threats related to the privacy issue. In addition, we further describe the fulfilled security requirements.

5.1. Physical and MITM attacks

An adversary may play with various technics of MITM attacks by initially impersonating as either legitimate RFID reader or legitimate RFID tag. In this case, an adversary may fool both devices to elicit their sensitive information. Thus, the adversary can intercept the key management system in order to reveal all security properties and further perform malicious activities (e.g. cloning attacks, tag emulating, collision attack, DoS, replay attack, etc.).

Nevertheless, an adversary will not be able to perform aforementioned attacks, as he is not able to find all critical parameters (i.e. *e, P, Q, g, H1* and *H2*) which are preloaded before the network deployment. Thus, an attacker will not be able to correctly calculate the initial session key and intercept or alter the messages in order to respond the mutual authentication procedure. Thus, all aforementioned threats can be mitigated by preventing an adversary to perform impersonation and various technics of MITM attacks.

Let us assume that an adversary tries to use different parameters (i.e. *e', P', Q', g', H1'* and *H2'*) in order to play with MITM. In this case, an adversary is not able to correctly respond the mutual authentication challenges by using the incorrect parameter. Moreover, the challenge is more complicated since an adversary has no chance to find the master secret key *s*, which is known only by the KGF. In this case, an adversary is not able to generate its correct pairs of public and private key affecting the messages cannot be properly encrypted and decrypted. Thus, both an adversary and the victim will not be able to connect to each other. Let us assume that the adversary use incorrect master secret key $k \neq s$ in order to impersonate as legitimate reader. The adversary then incorrectly generates its pair of public key $r_{Adv} \neq r$ and private key $R_{Adv} \neq R$:

$$r_{Adv} = H1'(\text{adversary MAC address}) \quad (5)$$

$$R_{Adv} = \frac{1}{k + r_{Adv}} P' \quad (6)$$

This case makes the legitimate RFID tag is not able to correctly calculate the initial session key *i* generated by the adversary, since:

$$i \neq H2'(e'(T, C1)) \oplus C2 \quad (7)$$

Moreover in the opposite way, the adversary is not able to correctly calculate the session key *j* generated by the legitimate RFID tag, since:

$$j \neq H2'(e'(R_{Adv}, C3)) \oplus C4 \quad (8)$$

Thus, an adversary is not able to access the information both from the legitimate RFID reader and legitimate RFID tag.

Furthermore, an adversary has no chance to play with MITM (i.e. alter the message) since the adversary is not able to elicit the encrypted message based on incorrect private key. In addition, an adversary will not be able to transit or forward the message since he is not able to find both the MAC address of legitimate RFID tag *t* and reader *r*, based on the incorrect parameters, since.

$$t \neq H1'(\text{tag MAC address}) \quad (9)$$

$$r \neq H1'(\text{reader MAC address}) \quad (10)$$

An adversary may inquiry the reader and tag MAC address from social engineering. Nevertheless, the adversary is still not able to generate the correct public key for the tag and the reader as well as their corresponding private key in order to play with MITM. Moreover, the adversary will not be able to respond the chippertexts based on the incorrect parameters. Let us assume that the adversary generate incorrect public key for the tag $t' \neq t$, public key for the reader $r' \neq r$, as well as incorrect chippertexts $C1' \neq C1$, $C2' \neq C2$, $C3' \neq C3$ and $C4' \neq C4$.

$$t' = H1'(\text{tag MAC address}) \quad (11)$$

$$r' = H1'(\text{reader MAC address}) \quad (12)$$

$$C1' = x(Q' + t'P') \quad (13)$$

$$C2' = i \oplus H2'(g'^{x}) \quad (14)$$

$$C3' = x(Q' + r'P') \quad (15)$$

$$C4' = j \oplus H2'(g'^{x}) \quad (16)$$

In this case, both the reader and the tag will not be able to respond the message and recover the session key based on incorrect chippertexts since:

$$i \neq H2(e(T, C1')) \oplus C2' \quad (17)$$

and

$$j \neq H2(e(R, C3')) \oplus C4' \quad (18)$$

Hence, both the tag and the reader will discard the connection since they cannot verify the message.

An adversary may try to conduct physical attacks by stealing the unsupervised RFID tag and copying its MAC address and all security parameters (i.e. *e, P, Q, g, H1* and *H2*). Nevertheless, an adversary is still not able to respond the mutual authentication challenge, since the master secret key *s* is only known by the KGF. In other word, it is never shared to any device neither to the tag nor to the reader. Thus, the adversary will not be able to correctly generate the pair of private key for the reader and the tag. Let us assume that the adversary use incorrect master secret key $k \neq s$. Here, we take example scenario from the tag side, in this scenario the adversary may be able to generate the correct public key for the tag. Nevertheless, the adversary incorrectly generates the tag private key $T_{Adv} \neq T$ based on incorrect master secret key *k*:

$$T_{Adv} = \frac{1}{k+t_{Adv}} P \tag{19}$$

Hence, the adversary is not able to respond the mutual authentication challenges, since:

$$\begin{aligned} e(T_{Adv}, C1) &= e(T_{Adv}, w(Q + tP) \\ &= e\left(\frac{1}{k+t} P, w(sP + tP)\right) \\ &= e(P, (sP + tP))^{\frac{w}{k+t}} \\ &= e(P,P)^{\frac{w(s+t)}{k+t}} = g^{\frac{w(s+t)}{k+t}} \end{aligned} \tag{20}$$

In this case, the initial session key generated by the legitimate reader is not equals as equation (1):

$$i \neq H2(g^{\frac{w(s+t)}{k+t}}) \oplus C2 \tag{21}$$

The case is same when the adversary tries to intercept message from the tag to the reader, affecting the adversary will not be able to intercept and alter the messages from both direction (i.e. the reader and the tag).

5.2. Privacy Threats

As typically in key management process of wireless communication, an adversary has chance to reveal the client privacy. In this regards, the adversary may reveal the RFID tag identity and exploit it to perform various activities, which are listed as follows.

- An adversary can reveal the tag location as well as conduct unauthorized tracking.
- The reveled identity makes an adversary has chance to perform various attacks including resources consumption attacks. It is performed by repeatedly sending amount of packets to drain the battery and wasting the limited bandwidth.

Nevertheless, our solution protects the identity from being revealed by unauthorized party by attaching it in the chippertext. In this case, the identity is firstly hashed to a 128 bit integer *t = H1(tag MAC address)* or *r = H1(reader MAC address)* before sending it to the intended recipient. Moreover, the protection is manifold improved since the hash function is included in the encrypted messages. In this regards, it is included in encrypted payload *ηT(C1, C2)* for *t* as destination address or *ηT(C3, C4)*for *r* as destination address. Thus, the privacy is preserved by protecting the identity in layered way (i.e. by firstly hashing the identity and by including it in the encrypted payload).

5.3. The Fulfilled Security Requirements

This section discusses the security requirements that have described in the first section. The following list outlines such requirements that are fulfilled by our key management solution.

- *Mutual authentication and authorization*. Our key management solution provides such feature, in order to prevent various threats on impersonation, fraudulence and various techniques of MITM (e.g. intercept the key management system, DoS, replay attack, etc.). In this case, the tag and the reader exchange the challenge in mutual way. Particularly, they exchange and verify the random value of (*C1, C2*) and (*C3, C4*) that are generated for each other. Indeed, if the exchanged two values cannot be verified, the key management process is aborted. Thus, it ensures that only authorized party can get access to the security service.
- *Availability*. Our solution ensures that the key management system is always available every time the RFID tag finds the located reader. In this regards, the RFID reader is always connected to the RFID database. In other word, it arguably ensures that the service is available whenever needed.
- *Privacy*. The identity is included in the encrypted payload (i.e. *ηT(C1, C2)* and *ηT(C3, C4)*), thus only corresponding recipient can find out the correct source or destination address. In addition, the protection is layered by firstly hashing the identity (i.e. *t = H1(tag MAC address)* and *r = H1(reader MAC address)*) before attaching it in the encrypted message.
- *Credibility*. It was proved on the previous section that our solution can ensure that all communication payloads are not being altered or transited, since only authorized party can get access to the system. In this regards, we proved that even an adversary who is able to get the security parameter (e.g. by performing physical attack to unsupervised RFID tag), he or she somehow will not be able to compromise the RFID communications.
- *Security strength and resistance*. Our solution relies on 128 bit security strength of *ηT* paring, which is about same as the 3072 bit of RSA method. Thus, it is strong enough to prevent the key management solution from various techniques

of brute-force attacks. Furthermore, it was demonstrated in previous section that the key management system can mitigate various security and privacy threats on smart RFID system, including cloning attacks, impersonation, resources consumption attacks as well as various technic of MITM attacks.

- *Communication overhead*. As described in section III, the key management systems only need to exchange the mutual authentication message in three-way handshake. The first message is 48 byte in length including 128 bit *C1*, 128 bit *C2* and 128 bit public key of the reader *r* (see figure 1). The second message is 64 Byte in length, including the four ciphertexts (i.e. *C1*, *C2*, *C3* and *C4*). Indeed, the maximum size of the authentication message that is required to be transported is only 160 Bytes in length. It is the third messages in the mutual authentication process, which includes 128 byte of the new private key and the two values of *C3* and *C4* with 128 bit each. Thus, it is doable to be transmitted in one frame of standard IEEE 802.15.4.
- *Computation overhead*. Our experiment demonstrated that the performance of our key management solution is affordable for the limited CPU options. It is even feasible for the lower option of clock frequencies at 104 MHz, which is done only in 0.057 sec.
- *Storage overhead*. The security properties that are stored in the RFID tag memory are *E* (1 Byte), *P* (128 Bytes), *Q* (128 Bytes), *g* (768 Bytes) and one private key *T* (128 Byte). In overall, the RFID tag requires a storage space less than 2KB.

6. Conclusion

Providing key management solution in smart RFID system is a complex challenges. However, we proved that our solution suits to the security requirements in the constrained nature of smart RFID system. It is affordable for the RFID tag that associates with limited resources (i.e. limited data-rate, limited CPU power, limited battery and memory storage). Furthermore, the analysis results presented that the key management solution can mitigate various threats including various technics of MITM attacks as well as various threats arisen from the use of RFID technic itself (i.e. cloning attacks, unauthorized tracking, impersonation, etc.). In addition, both the RFID tag and the reader identities are protected in layered way. It is conducted by firstly hashing it in 128 bit integer and sequentially attaching it in to the encrypted payload. Thus, the privacy can be preserved and all possible threats arising from the revealed identity can be mitigated as well.

References

[1] M.F. Sadikin, M. Kyas, "RFID-tate: Efficient security and privacy protection for active RFID over IEEE 802.15.4," The 5th International Conference on Information, Intelligence, Systems and Applications, IISA 2014.

[2] J.-L. Beuchat, E. Lpez-Trejo, L. Martnez-Ramos, S. Mitsunari, F. Rodrguez-Henrquez, "Multi-core implementation of the tate pairing over supersingular elliptic curves," Proceedings of the 8th International Conference in Cryptology and Network Security, December, 2009.

[3] M.F. Sadikin, M. Kyas, "Security and Privacy Protocol for Emerging Smart RFID Applications," The 15th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2014.

[4] M.F. Sadikin, M. Kyas, "IMAKA-Tate: Secure and Efficient Privacy Preserving for Indoor Positioning Applications," The 5th Int. Conf. on Smart Communications in Network Technologies (SaCoNet), 2014.

[5] C. Mulkey, D. Kar, A. Katangur, "Towards an Efficient Protocol for Privacy and Authentication in Wireless Networks," The 12th International Conference on Security and Management, July 2013.

[6] P. Szczechowiak, M. Collier, Tinyibe: Identity-based encryption for heterogeneous sensor networks, in: 5th International Conference on Intelligent Sensors, Sensor Networks, and Information Processing (ISSNIP), 2009, pp. 319–354.

[7] X. Xiong, D. Wong, X. Deng, "Tinypairing: A fast and lightweight pairing-based cryptographic library for wireless sensor networks," IEEE Wireless Communications and Networking Conference (WCNC), 2010.

[8] Recommendation for key management - part 1: General, National Institute of Standards and Technology (NIST), available: http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57_part1_rev3_general.pdf? (last accessed: 12 February 2015).