

Effective
Efficient(?)

can overtake "privacy
ensuring" ?

LIGHTWEIGHT AUTHENTICATION PROTOCOL FOR LOW-COST RFID TAGS

P-F. BONNEFOI, P. DUSART

ABSTRACT. Providing security in usage of RFID tags is a challenging task due to their highly resource constrained nature. Thus, we have to rely on low-cost solution. Because tags cannot support strong cryptography, special lightweight algorithms and protocols need to be designed.

In this paper, we discuss some solutions and exhibit their deficiencies in order to design our own protocol. We propose an extremely lightweight mutual authentication protocol between a tag and a database sharing a common secret value. This protocol could be tailored to allow the database to disclose or not the identity of the tag to the reader by modifying the final steps of this protocol. We examine the adequacies of our solution regarding the chosen security requirements and its withstanding against some common and less common attacks.

low-cost ?
or
low cost

computation!

dedicated

inhibit

save ? / results?

pas clair

verbal
Strength?

bag.

additional ?

1. INTRODUCTION

The RFID (Radio Frequency IDentification) tags are a new kind of barcode with some additive properties. The main purpose of these tags is to provide remote identification (ID) by radio. Hence, many applications are developed to identify a product, an animal, a person using radio waves. For many reasons, these applications must be designed carefully in order to ensure privacy and security in the different area like stock management (or books management in library), fast shopping checkouts, pets identification, anti-counterfeiting, sensor or vehicular networks.

wireless

on parle
de RFID

The RFID tag takes place into an infrastructure composed of the tag itself, a reader and a database. This infrastructure defines the "RFID-system". The RFID tags need no internal battery which guarantees a long life-duration, and the communication distance is limited but sufficient for many applications (a tag could be activated up to 100 meters but could only communicate up to a few meters). For wide development, the tags must be constructed with low-cost electronic components which limit the computation capabilities and the storage space available. Due to cost reasons, the tags are not truly tamper-resistant, but provide sufficient protection that requires too expensive methods to disclose sensitive information like internal keys. But an adversary can abuse the communication channel in order to obtain some secret information by performing various attacks like traffic analysis, eavesdropping or more sophisticated attacks in order to disrupt these communications by disabling the tag or the infrastructure.

Each tag is associated to one unique identity. This identity could lead to a serious breach of privacy. This issue could be addressed by protecting this identity

Date: May 26, 2008 .
Keywords and phrases. RFID, Authentication.

1

power is supplied by the reader through an "induction" process. However

manufactured

reverse engineering hardware
privacy concerns.

dis citations!

do not need an
have

option!

deployment/
adoption

skill

designed

(A) since ~~extensive~~ power is supplied by the reader through an "induction" process. However

e.g.)
a n'importe quel support de
course à conséquence
je dirai:
with low cost
and low capability
for economic
reason

reader through

pluriel!

faire intro ?
pour comprendre le lien
ou dire ?

je ne comprends pas.
mutual (!)

not authorized
party
equipments
entity?

Introduire le plan + conclusion

P-F. BONNEFOI, P. DUSART

to be disclosed to improper material or individual. This protection is achieved by authenticating one interlocutor with the other one when a communication occurs between the tag and the related database through the reader.

RFID is a very active field for research, so we have to carefully position our work in regard to other proposals. We have to select the targeted tag hardware. Some of them provide hash function and are out of our scope, furthermore from [9] a study exhibits that hardware integration of hash function in a tag should be dismissed in favor of support of strong symmetric cryptography like AES[14]. In order to find a common way to classify tag hardware we will use an already presented and useful one in the following section. Next we will define our expected security goals. We will describe some common attacks. Moreover we will present an attack transposed from other secure devices like smart card and show that this attack has to be taken into account. Next we will detail our protocol. Finally, we show that our protocol fulfills the requirements and seems to be practical with the real world's constraints.

Tag? Classification?

lesquels?
dans objectifs dans goals. Classification goals/requirements

between the Tag and

2. HARDWARE FEATURES AND SECURITY GOALS

2.1. Infrastructure efficiency. As we have written before, RFID tags take place into an infrastructure, where a database is the central element. Within this infrastructure, we will define the SID, Secure Identifier, of which the knowledge is only shared with the database. We must ensure that information related to the tag could be retrieved without any uncertainty from the database and with time constraints. This issue is not easily resolved when it is combined with privacy and security concerns.

2.2. Hardware concerns. Chien [5] proposed a tag classification divided in two families:

- the first one is named "high-cost" with a microprocessor. There are tamper-resistant (up to AVA-VAN4 in Common Criteria[6]) and support some cryptographic functions (full-fledged support classical ones like 3DES, SHA-1, RSA and "Simple" are limited to random number generators and one-way hash functions).
- the second one is named "low-cost" without a microprocessor. There are not resistant to physical attacks. The "lightweight" tags support only a random number generation and simple functions like a Cyclic Redundancy Code (CRC) checksum, but not a cryptographic hash function. The "Ultralightweight" tags can only compute simple bitwise operations like XOR, AND, OR, etc.

phor des "

In our proposal we assume that ... and we will use ...

In our proposal, we will use a PRNG (Pseudo Random Number Generator) because we assume that we have a lightweight tag. With this tag, we want to ensure mutual authentication within the RFID infrastructure, between the tag and the database independently of the reader.

We have now to list the different requirements, constraints and goals of our proposals:

- lightweight: no possibility to use strong cryptographic functions. Hash functions are merely too expensive to use on low-cost tag [9].
- no traceability: identifying a tag for improper reasons could lead to trace any movement of this tag (using a rogue reader). So, the object tagged with a unique identity could be used to trace also its owner. In the case

à voir -

et pourquoi pas en utilisant 1 vrai reader?

il y a la sécurité ou pas ?

ou?
pour exemple on s'attend à ce que le soit section 3.

est ce vraiment son? (requirements?)
je ne comprends pas (toujours pas)

whose

VLA? finalement c'est VAN je crois.

c'est une bombe !!

Thus

of non unique identity for one tag, different tag's identities related the set of owned objects could be combined in order to define a profile that could trace their owner.

- privacy: first, we have to ensure no traceability and ^{then} second, we have to achieve anonymity by providing the choice to reveal or not the tag identity at the reader level. We see no use for anonymity at the database level, and we can ensure tag's identity protection based on SID: one common identity shared by the tags and the database exchanged securely through the reader, this reader being unable to get access to this identity. We can also apply the same method to share a secure group identity between different tags.
- resilience: the RFID system must be protected and its capacity to provide services must be ensured. This requirement leads to protect the infrastructure against some common attacks like DOS (Denial of Service).

3. COMMON ATTACKS ON RFID TAG

In this section, we will list some common attacks regarding the RFID infrastructure. Some of them will be illustrated by applying them on already proposed security enforcement protocols.

3.1. Attack on one participant

3.1.1. Attack on the tag. An attacker masquerades as a valid reader. Without knowing secrets, he tries to be authenticated. The goal is to extract sensitive informations. This attack generally is counteracted by a shared secret. The communications must also not reveal the identity of the tag or if it was ever seen or not (traceability).

3.1.2. Attack on the reader. Here the goal is to masquerade as a valid tag. The attack is not avoid by a shared secret because the attacker could replay recording values of a good authentication. This attacker is known as "Replay attack".

3.2. Attack on communication

3.2.1. Incompleteness. As a radio link is used, unintentional communication ending or transmission errors can happen. Hence the communication and the tag's work can be stop at any stage of the protocol.

3.2.2. Man-in-the-middle attack. This attack is a form of active eavesdropping in which the attacker makes independent connections with the victims and relays messages between them, making them believe that they are talking directly to each other over a private connection when in fact the entire conversation is controlled by the attacker. The attacker must be able to intercept all messages going between the two victims and inject new ones, which is straightforward in many circumstances, especially in wireless communications (for example, an attacker within reception range of an unencrypted Wi-Fi wireless access point, can insert himself as a man-in-the-middle). A man-in-the-middle attack can only be successful when the attacker can impersonate each endpoint to the satisfaction of the other.

3.3. Attack on Database

vous avez parlé d'1 identité / tag

je ne comprends pas

We do not think that --- is useful

1 identité, 1 participant, entrée & sortie

je ne comprends pas.

this method can also be used

from the tag

Phred?

des citations ---

before

by using --- in the next process

between Tag and ?

phred on re peut pas faire de replay on 3.1.1? protocol d'authentification?

pas sûr de la pertinence de l'exemple

Can est ds 4 RFID

il y avait 1 papier ds wikip 2004 il me semble -

des citations!!

3.3.1. *Desynchronization*: To authenticate a tag, generally database shares a secret with the tag. Due to low-cost constraints, the method is to use an initial secret that is being refreshed (a new value is derived from the previous) at the end of each successive communication. The security relies on the hiding of the function used to derive value and of the initial value: the new value could not be predictable from the disclosure of the previous one and each end of the communication are solely to know this function. An attacker has only to spoof a reader by sending a request to desynchronize the secret (one the end used the next or prior value of the other end). This attack is well known and can be found in [8]. For example, in [20], Vajda et al. study the following lightweight protocol:

Reader $\rightarrow$ Tag : $a^{(i)} = x^{(i)} \oplus k^{(i)}; k^{(i)} \oplus k^{(i-1)}$
 Tag $\rightarrow$ Reader : $b^{(i)} = x^{(i)} \oplus k^{(0)}$

where $i = 2; 3; \dots$ is a counter, which is increased by one in every new run of the protocol. If a fake reader asks a tag, the chain of secrets k is corrupted and all the future authentications will fail. This attack is taken into account in Gossamer protocol [15], where there are a chain of two keys in tag memory which are only updated in case of mutual authentication. The second key is a backup in case of abnormally end of the updating phase.

3.3.2. *Random authentication*. A fake tag can be authenticated by the other side in lucky cases. Usually, authentication of one tag is related to the finding of the SID of the tag in the database. After a successful authentication, data related to the tag could be retrieved according to this SID. Sometime, in ultra-lightweight protocols and due to time constraint, it could be interesting to combine authentication and index computation. So, these algorithms use some exchanged parameters in order to compute only the index of data related to the tag, and considers that, if this index exists in the database, then the tag is authenticated. In this case, the SID becomes a SIDI, Secret Identifier index.

For example in Chien et al.'s scheme [4], as we have written before, the tag is authenticated if a good identifier SIDI was found in the database. Let have a fake tag which computes nothing. At the first step, it answers two numbers R_2, R' which are random or fixed. The SIDI is never exchanged. Hence, the database tries to find a SIDI which satisfies $Left(Rotate(SIDI, \tilde{g}) \oplus \tilde{g}) = R'$ (the *Rotate* function is used to easily scramble the input). If the two numbers match, there is one SIDI which satisfies the equality. The fake tag needs only to accept the proposed challenge. This attack depends on the number of SIDI in the database: more SIDI are present in the database, easier it is to find a pair of R_2, R' that can match with an existing SIDI and lead to an incorrect secure acknowledge from the database. This attack could corrupt the whole RFID system but could not be easily reproduced, in order to make the fake identity permanent (because the authentication depends on a random value given by the database). Hence, a fake tag can be authenticated in $(database\ size / (2^L))$ cases, where L is the bit size of R' .

3.4. Attack on Implementation.

3.4.1. *Attack on values*. Here we suppose that we have a reader with a corrupted random number generator. The RNG sends always the same value (always zero). If the implementation doesn't verify this point, the nonces n_1, n_2 in Gossamer protocol [15] equals to zero. Then the reader computes the next values $(A, B, \dots)$

does not

an

finalemment je l'avais dit!

Toutefois on est de la partie sur l'attaque de l'ADBS et on parle que de la machine de la tag!

c'est quoi cet index?

oui et on ne s'en rend pas compte et on ne vérifie pas si l'authentification est réussie ou non!

il faut présenter le protocole et on ne comprend rien

what is, or, a, b.

ben oui c'est logique

think / believe guess/suppose

which could be c'est quoi?

il faut présenter le protocole.

où est R_2

defective

without errors. But with the null nonces, $n_3 = \text{MixBits}(0, 0) = 0$ and $n'_1 = \text{MixBits}(n_3, 0) = 0$. The reader sends correct values of A, B, C with $C = k_1^* + \pi + k_2^*$. The tag finds that the values of the reader are correct and sends D . But $D = k_2^* + ID + k_1^*$. The value of π is defined by the scheme, so an attacker with a modified reader can trace all tags who are presented by computing $ID = D - C + \pi$. In this case, with a corrupted reader, the traceability is possible.

3.4.2. Timing attack. Here we use a well-known attack on implementation of smart card: the timing attack. The timing attack is a side channel attack presented by Kocher [12] in which the attacker attempts to compromise a cryptosystem by analyzing the time taken to execute operations (like cryptographic algorithms). Every logical operation in a tag takes time to execute, and the time can differ based on the input; with precise measurements of the time for each operation, an attacker can work backwards to the input. This attack is very efficient when a secret is used for a loop bound. For example in a previously discussed algorithm, we need to compute $\text{Rotate}(x, k)$ where x is a variable and k a secret. The function achieved the left rotation on x of k positions. This function is implemented with a loop on an elementary function: left shift of one position. While the number of rotations is not reached, the memory is rotated for one position. Hence with classical implementation, a computation of $\text{Rotate}(x, 3)$ takes 3 times of a $\text{Rotate}(x, 1)$ computation. The number of rotations can be found by code examination (generally not possible) or estimated time of a Shift. We can also spy on time of a few authentications and deduce the time for one rotation using the greater common divisor. For example, if you measure $45 \mu s$ and $60 \mu s$ for time of rotation part, a single rotation will be a divisor of $\text{gcd}(45, 60)$ and takes $1, 3, 5$ or $15 \mu s$ as $\text{gcd}(45, 60) = 15$. The good value will be found by considering another computation time or by considering the most probable value. Hence by recording time of rotate computation you can evaluate precisely the number of rotations of x . Then if the number of rotations is a secret, it can be found by Timing attack.

If you apply this method on the first step of Chien et al's protocol [4], you can find the number of rotations of SID (Secret IDentity) which is denoted by $\tilde{g}$. Hence by timing considerations, the value of $\tilde{g}$ is known. As R' can be recorded easily, the left part of SID is known. We rotate SID in the opposite way to find a part of the secret SID. Next by recording and listening several new transactions, an attacker can retrieve all bits of SID.

If we apply this method on Gossamer's protocol [15], in the step "Extract n_1 from A" the tag must use the secret keys k_1 and k_2 as loop bounds to make the inverse computation of the reader (in fact $\text{ROT}(A, -k_1)$). We are exactly in the presented case. But the ROT function chosen is computed $\text{mod } L$ hence we can obtain $k_1 \text{ mod } L$.

A timing attack is an example of an attack that exploits the implementation of an algorithm rather than the algorithm itself. The same algorithm can always be reimplemented in a way that leaks little or no information to a timing attack. consider an implementation in which every call to a subroutine always returns in exactly x seconds, where x is the maximum time it ever takes to execute the routine. In such an implementation, timing gives an attacker no useful information; it also has the adverse effect of slower response times on average.

But in case of tag, it is not realistic to take maximum time for protocol; As the number of loop is random and not limited (only by memory representation), the

on other que (pour l'ajout, ...)

0 zero!

C pour 1 pour l'ajout

il faut passer le protocole.

Bf.

etc il faut
1 reader de faille
"vrai" et 1 modification
m question

?

il faut homogénéiser

??

afournir l'information

rotate?

for example,

Je

ne vois pas trop de lien entre les 2 phrases

A réaction

an attacker cannot get measurement by timing?

on max?

on ne peut pas décaler
faire
pas observable
sans

k_1 can be obtained mod L

a flow in the

execution time can be very long. It's bad for identification duration and for tag consumption.

4. PROTOCOL PROPOSAL

4.1. Assumptions. We assume that:

- the tag is not tamper-resistant but the technical requirements needed to recover the internal data provide sufficient security against the attacker.
- there exists a function named g which is lightweight and one-way (one can't retrieve the input thanks to the output $\tilde{g}$). We assume that this function is not sensible to timing attack. For this function, one can use the lightweight Hash-Function Tav-128 described in [16]. One can use also F-FCSR Stream Cipher [3] which is at the upper bound lightweight class (the implementation requires 4.7K logic gates) and, unlike LFSR, resistant to linear and correlation attacks.

4.2. **Definition.** $S(X, n)$ is a circular shift of X by n positions to the left. *Left* and *Right* are two substring functions which give the left part constituted by the left $L/2$ bits of an L -bits number representation (respectively the right $L/2$ bits). Each tag stores a secret identifier (SID), the index (SIDi) of SID in the database and a key K in its memory.

5. PROTOCOL ANALYSIS

5.1. Security Analysis.

- Untraceability property:** We use the same approach than [7] to define the property of untraceability that could be expressed through the following rule: for two traces given by the processing of the protocol, it is not possible to determine if these two traces have been obtained by the same tag processing the protocol or by two different tags. For our protocol, the exchanged variables R_1, R_2, n_1, n_2 are random. The tag can be identified by SID or SIDi. The value SID is used as input of the g function. As g is considered as a one-way function, one can't retrieve the input thanks to the output $\tilde{g}$. The value SIDi masked with M is named SID' . As g is a random function, its output will look random to an adversary. Hence M is random and SID' is also random. So no identity-related information can be derived from this value. Only the Database knows the identity of Tag. An adversary has a probability of $1/2^{(2l+2k)}$ to identify a tag another time, where l is the length of R_i and k is the length of n_i . This probability is valid if the adversary only eavesdrops the communications. But if an attacker has a fake reader (he can begin the communication with a tag) wants to know if the same tag appears another time, he has previously stored a session and there are also two ways to identify the tag (through SID or SIDi). In this case, the attacker is in an active phase and he could choose the values R_1 and n_1 . To identify SIDi, he must find the same SID' . As g is one-way, the attacker can choose R_1, n_1 but not R_2, n_2 . To identify SID, he must first have the same R . But the same R doesn't imply to have the same $\tilde{g}$. Hence, the probability of identify another time the tag is

$$(\text{number of recorded sessions}) / 2^{\text{binary size}(R_2) + \text{binary size}(n_2)}$$

power?
tamper-resistant
(merci pour
ou source's)
si?

pourquoi ne pas
utiliser des
notations
if this is ok?

d'avant
lex: Rotate.

pratique!

notation

Database
(si on
reste cohérent
avec le msc)

phase on
voulant s'en
aller.

secondant
avec?

enfin...

extraire

can not

je ne suis
pas sûr qu'on
présente
cela comme
ça

donc ce qui
est unique à tag!

! pseudo

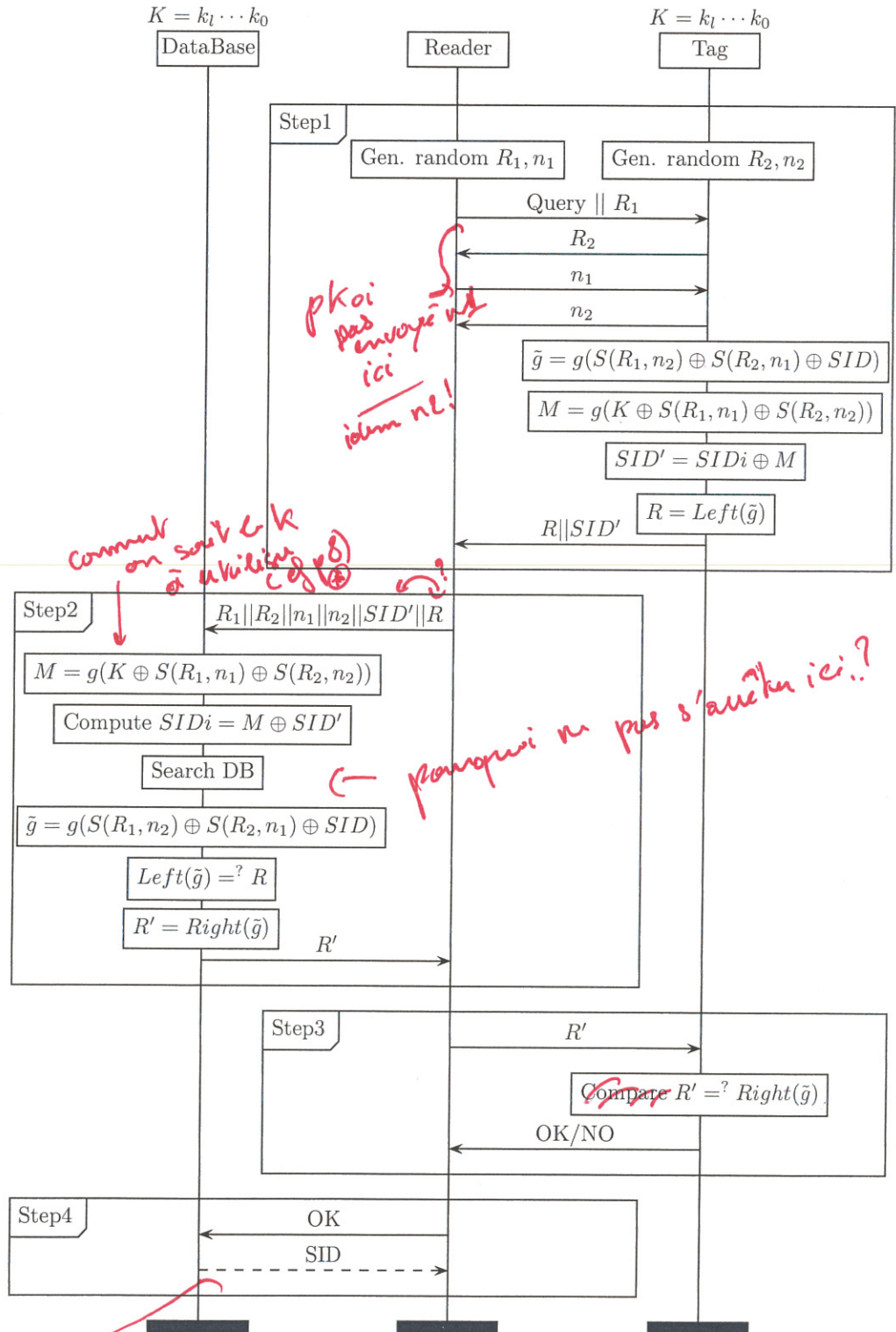
c'est quoi
Ridni
==

having

mettre en mode
math

enh!!

msc Our Protocol



présenter les étapes et le but.

pkoi pas envoyer ici idem ne!

comment on sait le k à utiliser (g-tilde)

pourquoi ne pas s'aider ici?!

optionnel
j' imagine!

mettre 1 nom et citer la figure

- **Mutual authentication:** The server identifies the tag by SIDI number, searches SID in database. Next the server authenticates Tag by comparing its computation (Left substring) with the tag challenge response.

The tag authenticates the server by verifying the server computation (Right substring). Since only the server and the tag have key K and SID as common secrets, the scheme provides mutual authentication. The SID can be used to sign the exchanges or in security operations.

- **Probability of random authentication:** The database authenticates the tag if $Left(\bar{g}) = R$. This equality depends on the behaviour of g function. The probability is : $P_1 = p(Entry = R) * p(K \oplus S(R_1, n_1) \oplus S(R_2, n_2) \in E_R)$ where E_R is the set $\{x / Left(g(x)) = R\}$. If we assume that S has a L -bits output, we have $P_1 \leq p(Entry = R) * (\max_R \#E_R) / 2^L$, which is negligible.

- **Replay attack prevention:** An Opponent can eavesdrop and record the communications between the tag and the reader. Next he can replay authentication with recorded data. As the values are random, the recorded data are not re-used another time. For this assertion, we suppose that we have a good random number generator.

- **Man-in-the-middle attack:** The proposed protocol doesn't protect to this attack [1]. This attack can be limited by using the anti-collision mechanism of RFID Tag and limit the communication distance. With these properties, only one tag will discuss with the reader and the distance between reader and tag can be reduced (1 or 2 feet). It became hard to be physically in the "middle" of the two devices as in WiFi based network.

- **DOS attack prevention:** The proposed protocol doesn't establish dependencies between each authentication. So each transaction is considered as a new one. This ensures that no synchronization is needed between one state of the database and one state of the tag. From the point of view of the tag, no one can put the protocol automata into a state from which it could not escape, due to the use of the special command query explained before. For the security of the database common solutions (like tracking queries of readers and blocking them in the the case of malevolent behavior) could be used to avoid reader to overload the database query engine. Study of these solutions is out of the scope of this paper.

- **Attack on values:** One must test if the value M is not equal to zero because in this case, the SIDI is transmitted without mask.

- **Timing attack:** One can perform a timing attack on our protocol. One can retrieve the two upper bounds used for the loop in the shift/rotation function. The two values are n_1 and n_2 . These values are public and so there is no interest to find them by this way.

5.2. Performance Analysis. For RFID tags, we need to have a computational and storage cost as low as possible. The price of a tag depends on its capabilities which are correlated with the previous costs. The communication cost is also important but not for the same reasons: if the communication cost is heavy, the protocol is slow to execute and an user must wait to have its authentication response.

5.2.1. Computational cost. The tag needs to access to a RNG (Random Number Generator) to create the challenges R_i and to a PRNG (Pseudo-Random Number

So nous l'avez dit!

*

Si on entre la step 1 on peut se faire passer par le tag autant de fois qu'on veut après ? non ? c'est q'non.

prendre la place du reader c'est min comme défini p3.

? je ne comprends pas

? je ne comprends pas. c'est quoi?

oui mais qui le sauve ? comment le différencier ? donc bug.

OK

Generator) designed by g . The other functions (XOR, Shift, Substring part) are very classical ones. You can change the g function to a hash function but this functionality is not suitable (unavailable) on low-cost tags. We suppose that the g function have one property of hash function: you can't retrieve the input seed with the knowledge of the output.

Lightweight property: We use a PRNG which is lightweight (but not ultra-lightweight), the circular left shift S which is considered as a ultra-lightweight function in [15] and the *Left* and *Right* Functions which can be implemented with masked boolean AND and shift (Example on 8 bits : $\text{Right}(X) = X \text{ AND } 0x0F$, $\text{Left}(X) = S(X \text{ AND } 0xF0, 4)$) are also lightweight.

The final cost of our protocol depends on the cost of the g function. We must choose a function for which the computational cost will take a small part of the time for a full transaction in common case. As we will see in the next section, this execution time must not max out the limit of 200ms but we suppose that the authentication phase[18] must not exceed 20ms.

5.2.2. *Storage requirements*. The tag must store its own secrets: the tag identity, SID, of length l , its identification number or place in the database, SIDI , and a global secret key K of the same length l . To achieve the computations, it handles a variable of the same length l .

5.2.3. *Communication cost*. The size of exchanged messages are: for reader-to-tag communication, we have a challenge R of l -bits, a shift value of no more than $\log_2 l$ bits, an authentication response of $l/2$ -bits; for tag-to-reader communication, it's almost the same but one must add the index SIDI in the database and the final response (OK/NOK) of 1 bit.

Real-world application: how our protocol would perform in real world? From the technical guidelines given by the GSM Alliance, [11], the transaction duration for contactless application is expected to be less than 200ms for Japan, and less than 250ms for european countries. In the next paragraph we will try to evaluate the time needed to compute our protocol and to make the query to the database.

The reader and the tag use the communication technology ISO 14443 (for proximity reading) that provides a throughput of 106kbit/s. For example, we can evaluate the amount of data exchanged over the network with a size of 128 bits for the common parameter l , by the following computation:

- for the step 1: 1 bit for the *Query*, $2 * 128 = 256$ bits, for R_1 , R_2 , and $64 + 128 = 192$ bits for R and SIDI ;
- for the step 3: $64 + 1 = 65$ bits for R' and the *Acknowledge*;
- total amount of data exchanged between the tag and the reader : 546 bits (209 bits sent by the reader to the tag, and 337 bits sent by the tag to the reader).

The computational complexity exhibited to retrieve data from the database is very limited, as we get a collision free identifier (SIDI) to access to the record associated to the current tag. So, we can achieve a good response time without adding overhead with our secure protocol for the communication with the database.

According to the throughput given previously, exchanging data related to our protocol requires: $546/106000 = 6$ ms. We have to add to this value, the time required to send the whole data frames in which our data must be enclosed (needed

encore...

homogénéiser

partir des fct. utilisées pour

cannot

2 2 2

et même on fait quoi?

finalité elle est globale?

l byts? bits? pas clair

et n'est-ce pas?

in the best case

Sans les computations!!

for APDU, the underneath protocol used to transport application data), and some values used to provide error correction to these frames (bit parity).

et donc le vps total est-il ?

5.3. Comparison with other protocols.

	U-Map [17]	SASI[5]	Gossamer[15]	Our Proposal
Privacy and Anonymity	Yes	Yes	Yes	Yes
Mutual authentication	Yes	Yes	Yes	Yes
Desynchro Resistant	No	No	Yes	N/A (Yes)
Disclosure Attacks	No	No	partial (cf 3.4.1)	Yes
Timing Attacks Resistance	Yes	strong	partial (cf 3.4.2)	Yes
Classification	ULight	ULight	Light	Light

We compare our protocol with a family of Ultralightweight Mutual authentication protocols presented by Peris et al.[17], SASI[5] presented by Chien, and Gossamer[15]. We don't say that that Gossamer protocol is ultralightweight like in [15] because the MixBit function uses add function which we consider lightweight only.

6. CONCLUSION

From the study of the security problems of a RFID protocol, we design a more secure protocol which satisfies the same requirements in terms of technical constraints.

We have addressed the deficiencies that we have exposed without sacrificing the requirement of realistic response time to exchange with the database. This response time is crucial in order to use the proposed algorithm in real world, like needed within the active research area of NFC, Near Field Communication (for example, in the case of transportation which requires acknowledgment between a database keeping the subscriber information and the tag validating the identity of this subscriber, the time to make the mutual authentication is over constraint).

The protocol could be modified according to different goals:

- if we keep only steps 1 & 2, authentication of the tag has been achieved from the point of view of the database. This could be sufficient for some applications where mutual authentication is not required like stock management; the database learned that this particular tag has been read and could carry some internal operations (remove from the current stock for example).
- if we keep steps 1, 2 & 3, mutual authentication has been achieved. The secret identity of the tag (SIDI) is not revealed to the reader.
- the step 4, used to disclose the tag identity to the reader, could be optional. The disclosure of this identity could be made according to the choice of the database only, or by the value of a flag parameter in the RFID key (one bit of the key could be used to set this choice). We can allow this behavior: if a direct and authorised connection between the reader and the database could be established, trust between reader and the tag is enabled and the tag allows the reader to know its identity. So, our protocol could rely on some more powerful secure protocol established between the reader and the database.

- this protocol could also be used to provide a secure communication link between tag and database, or tag and reader enabling some updates of the tag's memory.

REFERENCES

- [1] ROSS ANDERSON, "RFID and the Middleman", <http://www.cl.cam.ac.uk/~rja14/Papers/rfid-fc07.pdf>
- [2] GILDAS AVOINE, RFID Security: Techniques, Protocols and System-On-Chip Design, chapter Scalability Issues in Large-Scale Systems. Springer-Verlag, 2008.
- [3] F. ARNAULT, T. P. BERGER AND M. MINIER, "On the security of FCSR-based pseudorandom generators", SASC 2007, The state of the art of stream ciphers, January 31-February 1, 2007, Bochum, Germany.
- [4] HUNG-YU CHIEN, CHEN-WEI HUANG, "A Lightweight RFID Protocol Using Substring", Lecture Notes in Computer Science Volume 4808/2007, Book Embedded and Ubiquitous Computing, Pages 422-431, DOI 10.1007/978-3-540-77092-3
- [5] HUNG-YU CHIEN, "SASI: A New Ultralightweight RFID Authentication Protocol Providing Strong Authentication and Strong Integrity," IEEE Transactions on Dependable and Secure Computing, vol. 4, no. 4, pp. 337-340, Oct.-Dec. 2007, doi:10.1109/TDSC.2007.70226
- [6] COMMON CRITERIA, "Part 3: Security assurance components", 2009, <http://www.commoncriteriaportal.org/files/ccfiles/CCPART3V3.1R3.pdf>
- [7] T. VAN DEURSEN, S. MAUW, AND S. RADOMIROVIC, "Untraceability of RFID protocols", In Information Security Theory and Practices. Smart Devices, Convergence and Next Generation Networks, volume 5019 of Lecture Notes in Computer Science, pages 1-15, Seville, Spain, 2008. Springer
- [8] T. DIMITRIOU, "A lightweight RFID protocol to protect against traceability and cloning attacks". In: Conference on Security and Privacy for Emerging Areas in Communication Networks SecureComm, Athens, Greece, IEEE (September 2005)
- [9] MARTIN FELDHOFFER AND CHRISTIAN RECHBERGER, "A Case Against Currently Used Hash Functions in RFID Protocols", in the proceedings of On the Move to Meaningful Internet Systems, Lecture Notes in Computer Science, Springer, 2006, pp. 372 - 381.
- [10] T. Good and M. Benaissa, "Hardware results for selected stream cipher candidates", 2007, <http://www.ecrypt.eu.org/stream/papersdir/2007/023.pdf>.
- [11] GSMA, "Mobile NFC Technical Guidelines, vs 2.0", november 2007, <http://gsmworld.com/newsroom/document-library/technical-documents.htm>
- [12] P. KOCHER, "Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems", CRYPTO 1996, pp. 104-113, <http://www.cryptography.com/timingattack/paper.html>
- [13] C. MAXFIELD and A. BROWN, "DIY Calculator Demo: Pseudo-Random Number Generator", 2005, <http://www.diycalculator.com/docs/demo-lfsr.pdf>
- [14] NIST, "Advanced Encryption Standard", <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>
- [15] PEDRO PERIS-LOPEZ, JULIO CESAR HERNANDEZ-CASTRO, JUAN M. E. TAPIADOR, AND ARTURO RIBAGORDA, "Advances in Ultralightweight Cryptography for Low-cost RFID Tags: Gossamer Protocol", <http://www.lightweightcryptography.com/Gossamer.pdf>
- [16] PEDRO PERIS-LOPEZ, "Lightweight Cryptography in Radio Frequency Identification (RFID) Systems", PhD thesis, October 2008, <http://pperis.host22.com/pperis/thesis/peris.thesis.pdf>
- [17] PERIS-LOPEZ, P., HERNANDEZ-CASTRO, J.C., ESTEVEZ-TAPIADOR, J.M., RIBAGORDA, A., "EMAP: An efficient mutual authentication protocol for low-cost RFID tags", In Meersman, R., Tari, Z., Herrero, P. (eds.) OTM 2006 Workshops. LNCS, vol. 4277, pp. 352361. Springer, Heidelberg (2006)
- [18] POUPARD G., STERN J., "Security analysis of a practical on the fly authentication and signature generation", EUROCRYPT '98, International conference on the theory and application of cryptographic techniques, 1998, vol. 1403, pp. 422-436.
- [19] B. SCHNEIER, "Applied Cryptography", John Wiley and Sons, 1996.

Confidentiality

multiple log policies; -)

- [20] I. VAJDA AND L. BUTTYN, "Lightweight Authentication Protocols for Low-Cost RFID Tags", 2nd Workshop on Security in Ubiquitous Computing, in conjunction with Ubicomp 2003, Seattle, Washington, USA, October 12, 2003.

XLIM - UMR CNRS n°6172, FACULTÉ DES SCIENCES ET TECHNIQUES, UNIVERSITÉ DE LIMOGES, 87060, FRANCE

Current address: Département de Mathématiques, 123 avenue Albert THOMAS, 87060 LIMOGES CEDEX, FRANCE

E-mail address: bonnefoi@unilim.fr, pierre.dusart@unilim.fr

qui a écrit

Désolé du retard. 2 ans !!