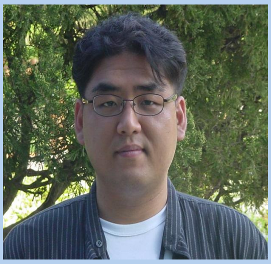
Toward Designing Provably Secure Cryptographic Protocols for RFID Tags

Dang Nguyen Duc (ICU),
Hyunrok Lee (ICU),
Kwangjo Kim (ICU)

Auto-ID Labs White Paper WP-HARDWARE-049

March 2009

Dang Nguyen Duc
PhD Student,
KAIST-ICC



Hyunrok Lee
PhD Student,
KAIST-ICC

Kwangjo Kim
Professor,
KAIST-ICC

Contact: nguyenduc@icu.ac.kr
CAIS Lab (R504), KAIST-ICC
119 Munjiro, Yuseong-gu, Daejeon, 305-732, Republic of Korea
Internet: http://caislab.icu.ac.kr/

Hardware

Index

Abstract

It is well known that RFID is subject to various security threats, most notably tag cloning and tracking. To cope with these security threats, we need to implement cryptographic protocols on RFID tags. However, designing a cryptographic protocol is a difficult process. It is even more difficult when the design is restricted by the limited computational power of the targeted devices. Meanwhile, RFID tag is perhaps the device with least computational power due to a very tight price constraint of a RFID tag. Therefore, designing a secure yet *lightweight* cryptographic protocol for RFID tags is both challenging and tempting. There are two approaches in designing cryptographic protocols for low cost and low computational power devices: finding more efficient implementation of existing protocols and designing new lightweight protocols from ground-up. This paper deals with the latter.

Underlying foundation for security of a cryptographic protocol is based on hard computational problem. Intuitively speaking, a cryptographic protocol is said to be secure if breaking security is computationally equal to solving a hard problem. Popular hard problems for existing cryptographic protocols include integer factoring (IP), discrete logarithm (DLP) and Diffie-Hellman problem (DHP). In this whitepaper, we discuss the advantages of designing cryptographic protocols for RFID tags based on unconventional hard problems rather than IP, DLP or DHP. We show an example by presenting several lightweight cryptographic protocols based on a hard learning problem called *Learning Parity with Noise* problem (LNP).

Keywords: RFID, Security, Authentication, LPN Problem

1 Introduction

RFID (Radio Frequency Identification) is a very promising technology to realize many powerful applications for supply chain management. The key idea is to attach to each and every object with an inexpensive and wirelessly readable tag, namely RFID tag. Each RFID tag carries a unique string to serve as object identity so that this identity can be used as a pointer to detail information about the corresponding object. Indeed, this scheme does not only benefit supply chain management but also consumer‘s experience because it allows users to keep track of their own tagged items and be aware of their surrounding environment filled with tagged objects. Ironically, this very idea of RFID also causes security concerns which could harm RFID adoption. These concerns are two-fold.

- *Counterfeiting Product*: Once RFID is widely used, we will become dependent on RFID to recognize surrounding objects, especially merchandise objects. However, the object identity stored in RFID tag can be read by any compatible RFID reader, not to mention wireless communication is inherently insecure. As a result, RFID tag can be duplicated and placed on counterfeiting products without being detected.

- *Costumer Privacy*: The uniqueness and availability of object identity poses another problem for end users, their privacy. With the vision of RFID tags being everywhere, it is not surprise a human would carry with him/her several RFID-tagged objects. Therefore, a malicious hacker equipped with a compatible RFID reader, can know objects the person carries as well as track the person's movement. That is only a simple scenario. Skillful hacker can be more sophisticated and do more damage.

To address the above security threats, a common solution is to implement cryptographic protocols between an RFID tag and an RFID reader such that they can be mutually authenticated and at the same time there is a privacy-preserving mechanism incorporated. Indeed, authentication and privacy-preserving are two well-studied issues in cryptography and there are many protocols known in literature. Unfortunately, these protocols cannot be used for RFID low-cost tags in a foreseeable future. The reason is those protocols require computational intensive operations including exponentiation, elliptic curve scalar multiplication, pairing and even block cipher. These operations are either beyond capability of current generation of RFID tags or heavily power-consuming for next-generation RFID tags. The first approach one can consider is to seek *lightweight* implementation of existing cryptographic protocols so that they can be implemented on RFID tags. However, there are several shortcomings in this approach:

- First of all, it is not easy and even impossible in case of extremely low-cost devices like passive RFID tags. In addition, there might certain limitation to how much one can improve performance of a conventional cryptographic protocol.
- Secondly, if lightweight implementation is possible; such implementation might be subject to attacks like side-channel attack (including timing attack, power analysis, *etc*) which can compromise security of the protocols.

The second approach is to design new cryptographic protocols with efficiency constraint in mind. Although the second approach might be more challenging, it is arguably preferable as it allows designers to aim for a good balance between security and efficiency from the beginning.

An important (and probably the most difficult) task in the design of a cryptographic protocol is to prove that the protocol achieves its desired level of security. A common method to prove security is to reduce breaking security to solving a hard problem. More specifically, a protocol is said to be secure, *i.e.*, satisfying its security goal, if breaking security is computationally equivalent to solving a hard problem. The most well-known hard problems in cryptography are number theoretic problems including integer factoring (IP), discrete logarithm (DLP) and Diffie-Hellman problem (DHP). The choice of hard problem to design a cryptogrphic protocol will dictate the minimal performance requirement of targeted devices. For instance, a protocol based on DLP will require the device to perfome exponentiation operation comfortably. We think that the first important step to design a provably secure and lighweight cryptographic protocol is to look for a suitable computational hard problem. A candiate hard problem should satisfy the following criterias:

- It should be well-studied and well-known to be hard, *e.g.*, a NP-hard problem.
- It should dictate lightweight operation.

In this paper, we present an example of designing lightweight cryptographic protocols based on an *unconventional* hard problem called *Learning Parity with Noise* (LPN) problem. We first describe a seminal work by Juels and Weis [12] in which they presented a RFID authentication protocol based on LPN problem called HB+. We then present our works including an encryption scheme and a key exchange protocol based on LPN problem.

2 LPN Problem

The LPN problem involves binary inner product of two *k*-bit numbers. The operation is defined as follows: given two *k*-bit number $a = (a_0a_1...a_{k-1})_2$ and $x = (x_0x_1...x_{k-1})_2$, the binary inner product of *a* and *x*, denoted as $a \cdot x$ is computed as follows: $a \cdot x = (a_0 \wedge x_0) \oplus (a_1 \wedge x_1) \oplus ... \oplus (a_{k-1} \wedge x_{k-1})$. Clearly, this operation can be easily implemented in cheap hardware. Furthermore, as noted by Juels and Weis [14], there is no need to buffer all *k* bits of *a* and *x* at once when evaluating $a \cdot x$. Therefore, memory requirement for this operation is also very low. The first cryptographic significance of binary inner product is due to Goldreich and Levin [7]. They proved that $a \cdot x$ is unpredictable if only either *a* or *x* is given. This result was subsequently used to construct a secure pseudo-random number generator (though not practical).

The first cryptographic protocol based on binary inner product was introduced by Hopper and Blum [11]. They presented a human authentication protocol such that the human only needs to evaluate one binary inner product operation, and generate a random bit. The protocol is called HB and is shown to be provably secure under the assumption that so-called *Learning Parity with Noise* problem is intractable. To better illustrate the LPN problem, we now describe the HB protocol. In the HB protocol, the human (denoted as *H*, also called the prover) and a machine (denoted as *C*, also called the verifier) share a secret *x* of *k*-bit long. The protocol consists of several executions of a basic challenge-response protocol which is described in Fig. 1.

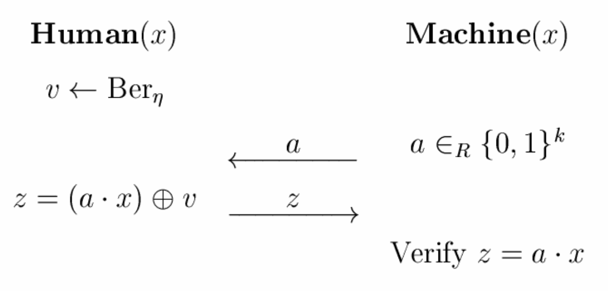


Fig. 1. One Round of HB Protocol

Ber_η denotes a Bernoulli distribution with expected value η where η is in (0, 0.5) range and called noise factor (that is the bit *v* - known as noise bit - is generated independently for each protocol round and equals 1 with probability η). The purpose of *v* is to prevent eavesdropping adversaries from extracting the secret *x* after observing *k* pairs (*a*, *z*). The machine accepts the human after, say *r* rounds of the above protocol if and only if human produces roughly *rη* incorrect responses.

It is straightforward that HB protocol is secure only if an eavesdropper observing messages exchanged between *H* and *C* has a negligible chance of impersonating *H*. More specifically, an eavesdropper *A* obtains *r* pairs (*a*, *z*) and tries to deduce a *k*-bit number *x*' such that using *x*' to carry out HB protocol, *A* would get accepted by *C*. The problem of finding such *x*' is called *Learning Parity with Noise* problem (LPN). However, as noted by Katz and Shin in [24], finding *x*' is essentially equivalent to finding *x* itself.

The LPN problem has been extensively studied in several research works including [8,9,10]. Those results show that LPN problem is very likely an intractable problem. To solve LPN problem, the best known algorithm by Blum *et al.* has sub-exponential complexity of $2^{O(k/\log k)}$.

3 Cryptographic Protocols based on LPN Problem

3.1 HB+ RFID Authentication Protocol

Since HB is a very lightweight protocol, it is desirable to use it for low-cost devices as well. However, HB is not secure against active attack in which RFID reader can be malicious. As we cannot assume RFID reader is trusted, HB cannot be used for RFID authentication. Juels and Weis was first to solve this problem by presenting an improved protocol called HB+ [12]. The protocol is an augmented version of its ancestor and offers better security strength. In the HB+ protocol, two parties, a RFID tag and a RFID reader, share two *k*-bit secrets (*x*, *y*) and the noise factor *η*. Similar to HB, HB+ also repeats a basic protocol *r* times. But the basic protocol of HB+ is a 3-round protocol with the RFID tag sending its random blinding factor *b* to the RFID reader first. The role of *b* is to prevent malicious RFID readers from extracting secrets stored in tag's memory by repeatedly querying the tag with the same challenge *a*. In effect, the response *z* is now computed as $z = (a \cdot x) \oplus (b \cdot y) \oplus \gamma$.

A description of the basic protocol is given in Fig. 2.

$$
\begin{array}{ccc}
\textbf{Tag}(x,y) & & \textbf{Reader}(x,y) \\
\gamma \leftarrow \mathrm{Ber}_\eta & & \\
b \in_R \{0,1\}^k & \xrightarrow{\quad b \quad} & \\
 & \xleftarrow{\quad a \quad} & a \in_R \{0,1\}^k \\
z = (a \cdot x) \oplus (b \cdot y) \oplus \gamma & \xrightarrow{\quad z \quad} & \\
 & & \text{Verify } z = (a \cdot x) \oplus (b \cdot y)
\end{array}
$$

Fig. 2. One round of HB+ Protocol

3.2 A Lightweight Encryption Scheme based on LPN Problem

We now describe a lightweight encryption scheme based on LPN problem. First, we observe that, given the pair (a, z = ($a \cdot x$) $\oplus$ v) where x is secret and (a, z) constitutes an instance of the LPN problem, then (a, z) is an encrypted message of the noise bit v. As Katz and Shin showed in [17], (a, z) is a pseudo-random bit string. Therefore, the above encryption scheme is semantically secure under ciphertext-only attack.

This encryption scheme might be inefficient in term of ciphertext expansion (1 bit plaintext resulting in k+1 bits ciphertext), but it is very simple and provably secure. We will show in the following Section how to use this encryption algorithm in a key exchange protocol.

3.3 A Lightweight Key Exchange Protocol based on LPN Problem

Let's assume that two entities A and B wish to establish a common secret key over an insecure channel. Using the above encryption scheme, A can securely transport 1 bit to B and vice versa. Then, one bit of the shared key can be computed by XORing two communicated bits. However, this trivial protocol is not secure against replay attack. To prevent replay attack and other more complicated attacks, we must use nonce when transport key material as well as provide key confirmation. To do so, we borrow ideas of the HB+ authentication protocol [12]. Similar to HB+, two entities in our key exchange protocol also share two k-bit secrets, say x and y. The protocol proceeds as follows (An illustration of the protocol is also given in Fig. 2.):

- $A \rightarrow B$: A sends (a, z_A) to B where a in $\{0, 1\}^k$ and $z_A = (a \cdot x) \oplus v_A$ with v_A is a randomly chosen bit.
- $B \rightarrow A$: B replies with (b, z_B) such that b in $\{0, 1\}^k$ and $z_B = (b \cdot x) \oplus (a \cdot y) \oplus v_A \oplus v_B$ with v_B is a randomly chosen bit.
- $A \rightarrow B$: if $v_A = v_B$, A sends a key confirmation message $c = (a \cdot s_1) \oplus (b \cdot s_2)$. Otherwise, it sends $c = (b \cdot s_1) \oplus (a \cdot s_2)$.
- B: upon receiving c, B verifies that c either equals $(a \cdot s_1) \oplus (b \cdot s_2)$ or $(b \cdot s_1) \oplus (a \cdot s_2)$ if $v_A = v_B$ or otherwise, respectively.
- A, B: the two parties compute 1 bit of shared secret as $v_{AB} = v_A \oplus v_B$.

In the above protocol, at first A securely sends its contribution to shared secret, v_A, to B. B not only sends back its contribution v_B to A but also incorporates a and v_A into its message to prevent reflection attack. To prevent unknown key share attack, A needs to send a key confirmation message c. Indeed, the message c only provides key confirmation for B.

Therefore, when *A* and *B* exchange the next secret bit, they can change their roles so that this time *A* is the one to receive key confirmation.

Regarding the relation of our protocol with the LPN problem, we can see that a collection of the pair (a, z_A) forms an instance of LPN problem with the noise factor η = 0.5. Indeed, there is no restriction on the noise factor in our protocol which means v_A can be drawn from any probability distribution rather than Bernoulli or uniform distributions. This allows flexibility in implementing our protocol on low-cost hardware. Note that, this is different in the HB and HB+ authentication protocols, where the noise factor has to be fixed and strictly smaller than 0.5 and roughly about 1/8 in practice. However, since the LPN problem becomes harder as the noise factor gets close to 0.5, we can see that our protocol potentially offers better security strength than HB and HB+ protocols.

$$
\begin{array}{lcl}
\mathbf{A}(x,y) & & \mathbf{B}(x,y) \\
v_A \in_R \{0,1\} & & \\
a \in_R \{0,1\}^k & & \\
z_A = (a\cdot x)\oplus v_A & \xrightarrow{a,\, z_A} & \\
 & & v_A = (a\cdot x)\oplus z_A \\
 & & v_B \in_R \{0,1\} \\
 & & b \in_R \{0,1\}^k \\
 & \xleftarrow{b,\, z_B} & z_B = (b\cdot x)\oplus(a\cdot y)\oplus v_A\oplus v_B \\
v_B = (b\cdot x)\oplus(a\cdot y)\oplus v_A\oplus z_B & & \\
\text{if } v_A = v_B & & \\
\quad c = (a\cdot x)\oplus(b\cdot y) & & \\
\text{else} & & \\
\quad c = (a\cdot y)\oplus(b\cdot x) & & \\
 & \xrightarrow{c} & \\
 & & \text{if } v_A = v_B \\
 & & \quad \text{verify } c = (a\cdot x)\oplus(b\cdot y) \\
 & & \text{else} \\
 & & \quad \text{verify } c = (a\cdot y)\oplus(b\cdot x) \\
v_{AB} = v_A\oplus v_B & & v_{AB} = v_A\oplus v_B
\end{array}
$$

Fig. 3. Our Proposed Key Exchange Protocol

4 Conclusion and Future Work

In this paper, we suggest that it is crucial to design provably secure and lightweight cryptographic protocols by looking for suitable security foundation that is a computational hard problem. We present an example by showing three cryptographic protocols based a hard learning problem called LPN. We believe that this is a right way to go when designing cryptographic protocols for next generation RFID tags.

We would like to note that HB+ is not secure against man-in-the-middle attack as shown in [13]. Several attempts to fix the issue have failed. However, we think the problem can be solved by using our proposed encryption scheme. We will investigate this solution in our future work.

References

[1] EPCglobal Inc., http://www.epcglobalinc.org/.

[2] Ari Juels, ``Strenthening EPC Tag against Cloning'', *To Appear in the Proceedings of WiSe'05*.

[3] Ari Juels, ``RFID Security and Privacy: A Research Survey'', *To Appear in the Proceedings of IEEE JSAC'06*.

[4] Stephen Weis, ``Security and Privacy in Radio Frequency Identification Devices'', Master Thesis, Available at http://theory.lcs.mit.edu/~sweis/masters.pdf, May 2003.

[5] Gildas Avoine and Philippe Oechslin, ``A Scalable and Provably Secure Hash-Based RFID Protocol'', *In the Proceedings of Workshop on Pervasive Computing and Communications Security - PerSec'05*, March 2005.

[6] Nicholas Hopper and Manuel Blum, *A Secure Human-Computer Authentication Scheme*, Proceedings of ASIACRYPT'01, Bart Preneel (Ed.), Springer-Verlag, LNCS 2248, pp. 149—153, 2001.

[7] Oded Goldreich and L.A. Levin, *Hard-core Predicates for Any One-Way Function*, 21st ACM Symposium on Theory of Computation, pages 25--32, 1989.

[8] Johan Hastad, *Some Optimal Inapproximability Results*, Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing, ACM Press, pp. 1--10, May, 1997.

[9] Michael Kearns, *Efficient noise-tolerant learning from statistical queries*, Journal of ACM Volume 45, Issue 6, ACM Press, pp. 983--1006, November, 1998.

[10] Avir Blum, Adam Kalai and Hal Wasserman, *Noise-tolerant Learning, the Parity Problem, and the Statistical Query Model*, Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, ACM Press, pp. 435--440, 2000.

[11] Nicholas Hopper and Manuel Blum, *A Secure Human-Computer Authentication Scheme*, Proceedings of ASIACRYPT'01, Bart Preneel (Ed.), Springer-Verlag, LNCS 2248, pp. 149—153, 2001.

[12] Ari Juels and Stephen Weis, *Authenticating Pervasive Devices with Human Protocols*, Proceedings of CRYPTO'05, Victor Shoup (Ed.), Springer-Verlag, LNCS 3261, pp. 293—308, 2005.

[13] Henri Gilbert, Matthew Robshaw and Herv\'e Silbert, *An Active Attack Against HB+ - A Provably Secure Lightweight Authentication Protocol*, Available at http://eprint.iacr.org/2005/237.pdf.

[14] Jonathan Katz and Ji Sun Shin, *Parrallel and Concurrent Security of the HB and HB+ Protocols*, Available at http://eprint.iacr.org/2005/461.pdf.