A FORWARD PRIVATE PROTOCOL BASED ON PRNG AND LPN FOR LOW-COST RFID

Xiaolin Cao and Maire O'Neill
Centre for Secure Information Technologies (CSIT), Queen's University Belfast, Belfast, U.K.

Keywords: RFID, Authentication, Forward privacy, PRNG, LPN.

Abstract: Low-cost Radio Frequency IDentification (RFID) tags are extremely resource-constrained devices, therefore, difficult to defend against corruption attacks. Meanwhile, forward privacy considers how to preserve the privacy of compromised tags. The majority of existing authentication protocols uses cryptographic hash functions to preserve forward privacy under the random oracle model, but the expensive hardware cost of a cryptographic hash function exceeds the budget of low-cost tags. In this paper, a novel forward private protocol for low-cost RFID applications is proposed. It is composed of a pseudorandom number generator (PRNG) and a learning parity with noise (LPN) problem. In comparison to previous protocols, the proposed protocol achieves a high forward privacy level and requires a small hardware cost. The proofs of security, correctness and forward privacy for the proposed protocol are provided under the standard model.

1 INTRODUCTION

RFID technology is increasingly being used in access control, product tracking and logistics management automation industries. A typical RFID system is composed of a powerful reader $\mathcal{R}$ (including the backend database system) and a number of resource constrained tags $\{\mathcal{T}_i\}_{1\leq i\leq L}$. It is difficult to add security features to protect them from corruption attacks. Meanwhile, forward privacy considers the privacy preserving problem after a tag is compromised (Ohkubo, 2003). Thus, developing forward private authentication protocols for low-cost tags is a major security challenge for the future development of RFID technology.

Much research have been carried out into lightweight protocols based on the learning parity with noise (LPN) problem, such as HB (Hopper and Blum, 2001), and its variants (Juels and Weis, 2005; Katz and Shin, 2006; Gilbert et al., 2008). These protocols utilize the hamming weight of a Bernoulli noise vector to achieve authentication purpose. The operations in the LPN problem involve the calculation of binary inner products and random noise bit generation. Computing the binary inner product only requires bitwise AND and XOR operations that can be computed on the fly. Therefore the LPN problem is a hardware-friendly primitive, and very attractive to low-cost RFID security. But these LPN-based protocols do not focus on the forward privacy. Previous work by Cao and O'Neill (2011) has employed the LPN problem to protect the forward privacy in the F-HB protocol, in which the LPN is used to update the key chain, and a secure PRNG to implement a message authentication code to defend against the GRS-MIM attack (Gilbert et al., 2008) for the LPN problem. However, the proposed protocol in this paper uses the PRNG to update the key chain, uses the LPN problem to calculate the authenticators, and does not worry about GRS-MIM attack.

In this paper, we improve on previous work and propose a forward private authentication protocol, PLFP, suitable for low-cost RFID applications. Firstly, it is composed of a secure PRNG and a LPN problem, which can be efficiently implemented. Secondly, it is provable secure and forward private under the standard model rather than the random oracle model.

The rest of the paper is organized as follows. In section 2, the LPN problem is defined. In section 3, the RFID security and privacy model is defined. The proposed PLFP protocol and proof are described in section 4. A hardware cost analysis and comparison results are given in section 5. Section 6 concludes this paper.

Cao X. and O'Neill M..
A FORWARD PRIVATE PROTOCOL BASED ON PRNG AND LPN FOR LOW-COST RFID.
DOI: 10.5220/0003520802870292
In *Proceedings of the International Conference on Security and Cryptography* (SECRYPT-2011), pages 287-292
ISBN: 978-989-8425-71-3

2 LPN PROBLEM

Let Ber_η denote the Bernoulli distribution with parameter $\eta \in (0,1/2)$, i.e. a bit $v \leftarrow \mathrm{Ber}_\eta$ is such that $\Pr[v=1]=\eta$ and $\Pr[v=0]=1-\eta$, while an l-bit vector $v \leftarrow \mathrm{Ber}_{l,\eta}$ is such that each bit of v is drawn according to Ber_η. Let U_n denote an oracle that returns a uniformly random n-bit string. Let $\mathrm{Hwt}(v)$ denote the hamming weight of vector v. Let S be a random $(l \times k)$ binary matrix, let c be a random k-bit vector, let $\eta \in (0,1/2)$ be a noise parameter, and let v be a random l-bit vector distributed according to $\mathrm{Ber}_{l,\eta}$, such that $\mathrm{Hwt}(v) \leq \eta l$. The LPN problem is such that given S, η, and $a \leftarrow (S \cdot c) \oplus v$, find a k-bit vector y such that $\mathrm{Hwt}((S \cdot y) \oplus a) \leq \eta l$.

For a fixed k-bit string s, let $\pi_{s,\eta}$ denote the oracle returning an independent $(k+1)$-bit string according to the distribution:

$$\{(c,(s \cdot c) \oplus v) | c \in_R \{0,1\}^k, v \leftarrow \mathrm{Ber}_\eta\}. \quad (1)$$

And for a fixed $(l \times k)$ matrix S, let $\Pi_{S,\eta}$ be the oracle returning an independent $(k+l)$-bit string according to the distribution:

$$\{(c,(S \cdot c) \oplus v) | c \in_R \{0,1\}^k, v \leftarrow \mathrm{Ber}_{l,\eta}\}. \quad (2)$$

Lemma 1 (Katz and Shin, 2006). Assume there exists an algorithm A making q oracle queries, running in time t, and $|\Pr[A^{\pi_{s,\eta}}(1^k)=1] - \Pr[A^{U_{k+1}}(1^k)=1]| \geq \epsilon$, then there is an algorithm B making $O(q \cdot \epsilon^{-2} \log k)$ oracle queries, running in time $O(t \cdot k\epsilon^{-2} \log k)$, and such that

$$\Pr[B^{\pi_{s,\eta}}(1^k)=s | s \in_R \{0,1\}^k] \geq \epsilon/4. \quad (3)$$

Definition 2.1. The advantage of distinguishing the oracle $\Pi_{S,\eta}$ and U_{k+l} in time t, Adv_Π, is defined as follows:

$$|\Pr[A^{\Pi_{S,\eta}}(1^k)=1] - \Pr[A^{U_{k+l}}(1^k)=1]|_{\mathrm{MAX}} \quad (4)$$

where the maximum value is taken over all A running in time at most t with no more than q oracle queries.

In the following sections, we use $\mathrm{LPN}_\Pi(\cdot)$ to represent the LPN problem. Every instance of $\mathrm{LPN}_\Pi(\cdot)$ has three inputs (S, v, c) and one output (a), such that

$$a = \mathrm{LPN}_\Pi(S,v,c) = (S \cdot c) \oplus v. \quad (5)$$

Lemma 2. Assume there exists an algorithm A making q oracle queries, within time t, and such that $|\Pr[A^{\Pi_{S,\eta}}(1^k)=1] - \Pr[A^{U_{k+l}}(1^k)=1]| \geq \epsilon$. Let t_π be the time taken to calculate a $\pi_{s,\eta}$ instance. Then there is an algorithm B making $O(q)$ oracle queries, running in time $t + \frac{l(l-1)}{2} t_\pi$, and such that

$$|\Pr[B^{\pi_{s,\eta}}(1^k)=1] - \Pr[B^{U_{k+1}}(1^k)=1]| \geq \frac{\epsilon}{l}. \quad (6)$$

Proof. We use a hybrid argument technique. Let S' denote a $(l-j) \times k$ binary matrix. We first define the following hybrid distribution D_j with $j \in [0,l]$ as $(c, r, (S' \cdot c) \oplus v)$, where $c \in_R \{0,1\}^k$, $r \in_R \{0,1\}^j$, $v \leftarrow \mathrm{Ber}_{l,\eta}$.

Upon a $(k+1)$-bit input, B draws a random $j \in [0,l]$ to construct a $(k+l)$-bit input as A's input. When $j < l$, it also needs to draw a random $(l-j) \times k$ binary matrix S'. It is straightforward that when B's input complies with U_{k+1}, the $j \in [1,l]$; when B's input complies with $\pi_{s,\eta}$, the $j \in [0, l-1]$. The distribution of D_l is the same as U_{k+l}, and D_0 the same as $\Pi_{S,\eta}$. And B uses A's outputs as its outputs. Thus

$$\begin{aligned} &|\Pr[B^{\pi_{s,\eta}}(1^k)=1|] - \Pr[B^{U_{k+1}}(1^k)=1]| \\ &= \frac{1}{l}\left|\sum_{0,l-1}(A^{D_j}(1^k)=1) - \sum_{1,l}(A^{D_j}(1^k)=1)\right| \\ &= \frac{1}{l}|\Pr[A^{\Pi_{S,\eta}}(1^k)=1] - \Pr[A^{U_{k+l}}(1^k)=1]| \geq \frac{\epsilon}{l}. \end{aligned}$$

Defintion 2.2. The LPN problem $\mathrm{LPN}_\Pi(\cdot)$ is said to be (t,ϵ)-secure if there is no adversary can distinguish $\Pi_{S,\eta}$ from U_{k+l} with advantage $\mathrm{Adv}_\Pi(t) \geq \epsilon$.

Lemma 3. Let t_Π be the time taken to calculate a $\Pi_{S,\eta}$ instance, and $\mathrm{LPN}_\Pi(\cdot)$ be a (t,ϵ_Π)-secure LPN problem. Suppose there is at most one valid $\Pi_{S,\eta}$ instance such that $z_0 = \mathrm{LPN}_\Pi(S,v,c_0)$, and q invalid $\Pi_{S,\eta}$ instances such that $\{z_i \neq \mathrm{LPN}_\Pi(S,v,c_i)\}_{1 \leq i \leq q}$, then based on the information above, the probability of obtaining a valid $\Pi_{S,\eta}$ instance is upper-bounded by $\epsilon_\Pi + 2^{-l}$ within time at most $t + O(qt_\Pi)$:

$$\Pr[A(c) = \mathrm{LPN}_\Pi(S,v,c)] \leq \epsilon_\Pi + 2^{-l}. \quad (7)$$

Proof. We claim that if there is an adversary A able to guess a valid pair utilizing the above information with advantage at least ϵ_A in time at most t, then we can use A to construct an algorithm B to distinguish $\Pi_{S,\eta}$ and U_{k+k} with advantage at least $\epsilon_\Pi + 2^{-l}$ with time at most $t + O(qt_\Pi)$.

After the algorithm B is given access to an oracle returning a $(k+l)$-bit string X, B randomly draws q $(k+l)$-bit strings to form the input to A. When B's oracle is U_{k+l}, these strings do not need any modification. When B's oracle is $\Pi_{S,\eta}$, then B needs some time to make sure that the other q random string are not $\Pi_{S,\eta}$ instances. These operations can

consume time at most $O(qt_{\Pi})$.

When B's oracle is U_{k+l}, it is clear that A outputs a valid pair with probability of 2^{-l}, because this uniformly distributed strings provide no useful information. When B's oracle is $\Pi_{S,\eta}$, A outputs a valid pair with probability of ϵ_A. We let B output 1 when A outputs a valid pair, else return 0. Then $|\Pr[B^{\Pi_{S,\eta}}(1^k) = 1] - \Pr[B^{U_{k+m}}(1^k) = 1]| \geq \epsilon_A - 2^{-l}$. From Lemma 2, we thus we get $\epsilon_A \leq 2^{-l} + \epsilon_{\Pi}$, and conclude the proof.

Lemma 4. Let $\mathrm{LPN}_{\Pi}(\cdot)$ be a (t, ϵ_{Π})-secure LPN problem. Then the probability of obtaining a collision between function $\mathrm{LPN}_{\Pi}(S_0, v_0, \cdot)$ and $\mathrm{LPN}_{\Pi}(S_1, v_1, \cdot)$ is upper-bounded by $\epsilon_{\Pi} + 2^{-l}$ within time at most $t + O(t_{\Pi})$, such that

$$\Pr[\mathrm{LPN}_{\Pi}(S_0, v_0, \cdot) = \mathrm{LPN}_{\Pi}(S_1, v_1, \cdot)] \leq \epsilon_{\Pi} + 2^{-l}. \quad (8)$$

Proof. We claim that if there is an algorithm A able to produce a valid collision utilizing the above information with advantage at least ϵ_A in time at most t, then we can use A to construct an algorithm B to distinguish $\Pi_{S,\eta}$ and U_{k+l} with advantage at least $\epsilon_{\Pi} + 2^{-l}$ with time at most $t + O(t_{\Pi})$.

After algorithm B is given access to an oracle returning $(k + l)$-bit string X, B constructs a $\Pi_{S,\eta}$ instance X_R to form the input (X, X_R) to A. Let $X_{(k)}$ be X's first k-bit binary digit, and $X_{(l)}$ be X's last l-bit binary digit. The collision implies that $X_{(l)} = X_{R(l)}$. When B's oracle is U_{k+l}, then $\Pr[X_{(l)} = X_{R(l)}] \leq 2^{-l}$. When B's oracle is $\Pi_{S,\eta}$, then $\Pr[X_{(l)} = X_{R(l)}] \geq \epsilon_A$. When A finds a collision, A outputs '1', otherwise, '0' is returned. B uses A's output as its output. $|\Pr[B^{\Pi_{S,\eta}}(1^k) = 1] - \Pr[B^{U_{k+l}}(1^k) = 1]| \geq \epsilon_A - 2^{-l}$. So we get $\epsilon_A < 2^{-l} + \epsilon_{\Pi}$, and conclude the proof. We note that this is a very loose upper-bound for the collision probability.

3 RFID SECURITY DEFINITONS

In this paper, an adversary A is assumed to be a probabilistic polynomial algorithm that is allowed to perform oracle queries during attacks. Five oracles are defined as follows.

O_1: It invokes the reader $\mathcal{R}$ to start a new authentication and returns the challenge message c.

O_2: It invokes tag $\mathcal{T}$ to responds with the response message a to challenge message c.

O_3: It returns intercepted and modified challenge, c, and response messages, a.

O_4: It returns the final authentication result.

O_5: It returns the current key and internal state information of a tag $\mathcal{T}$, and also updates the key and state information of tag $\mathcal{T}$ if necessary.

Definition 3.1 (q, t)-adversary. An adversary with his running time upper-bounded by t and the ability to disturb at most q authentication exchanges in this interval is called a (q, t)-adversary.

Definition 3.2 Security. The authentication protocol is said to be (q, t, ϵ)-secure if there exists no (q, t) -adversary who can impersonate an uncorrupted with the probability more than ϵ.

Definition 3.3 Correctness. The authentication protocol is said to be (q, t, ϵ)-correct if there is no (q, t)-adversary who can make an uncorrupted tag being rejected in an undisturbed authentication with the probability more than ϵ.

The forward privacy experiment is a two phase experiment. During the 1st phase, adversary A disturbs $r \in_R [0, q]$ authentications with oracle $(O_i)_{1 \leq i \leq 5}$, and outputs two uncorrupted challenge tags $\{\mathcal{T}_0, \mathcal{T}_1\}$. On entering the 2nd phase, the experiment chooses a random bit b, and b is concealed from A. Then A continues to disturbs with oracle $(O_i)_{1 \leq i \leq 4}$. Aftern A is given access to the O_5 oracle query to obtain the internal states of $\mathcal{T}_b$ and $\mathcal{T}_{1-b}$, A is no longer able to access any oracle related to $\mathcal{T}_b$ and $\mathcal{T}_{1-b}$, but A can access any other oracles. Eventually, A is asked to guess the random bit b by outputting a bit b'.

Definition 3.4. The advantage of a (q, t) -adversary A in the above experiment is defined as:

$$\mathrm{Adv}_A^{\mathrm{ForPriv}} = |\Pr[A \text{ outputs } 1] - 1/2|. \quad (9)$$

Definition 3.5 Forward Privacy. An authentication protocol is said to be (q, t, ϵ) -forward-private if there exists no (q, t)-adversary who can make the $\mathrm{Adv}_A^{\mathrm{ForPriv}}$ more than ϵ.

4 PROPOSED PROTOCOL

The proposed PLFP protocol is depicted in Figure 1. On each tag $\mathcal{T}$, there is a (t, ϵ_g)-secure PRNG $g: \{0,1\}^m \to \{0,1\}^n$ with computation time t_g as defined in Berbain et al. (2009), an LPN problem $\mathrm{LPN}_{\Pi}(\cdot)$, and a void function $\mathrm{GenSec}(\cdot)$, which is used to derive a secret matrix S of the LPN problem from the secret state σ of the PRNG. During the initialization stage, the tag $\mathcal{T}$ shares its initial internal state, a randomly m-bit secret σ_0, with the reader. The PLFP protocol runs as follows:

1. The reader $\mathcal{R}$ sends a challenge $c \in \{0,1\}^k$ to the tag $\mathcal{T}$.

2. The tag $\mathcal{T}$ first derives a matrix S from its current state σ using $S \leftarrow \text{GenSec}(\sigma)$, and a noise vector $\upsilon \leftarrow \text{Ber}_{l,\eta}$, where σ is derived using PRNG g in the previous transactions from σ_0. Secondly, it updates internal state $\sigma \leftarrow g(\sigma)$. Finally, the tag answers $a \leftarrow \text{LPN}_{\Pi}(S, \upsilon, c)$.
3. Upon receiving the response a, the reader $\mathcal{R}$ verifies the response by searching tags state chains: for each tag $\mathcal{T}$ in the system, the reader uses its last authenticated state σ_{cur} to run a sliding window length ω to look for a state σ_{cur+i} satisfying $a \oplus (S \cdot c) \leq \eta l$. If such a match is found, the reader accepts the tag $\mathcal{T}$ and updates its view of the tag's state by the successful match state $\sigma_{cur} \leftarrow \sigma_{cur+i}$. Otherwise the reader rejects the tag and does not update its view.

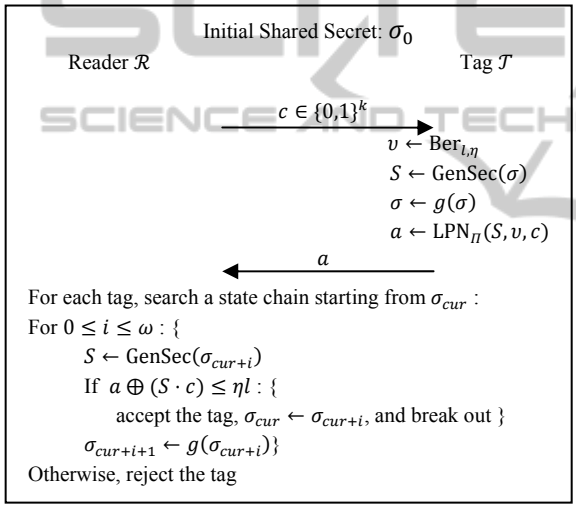

Figure 1: The proposed PLFP protocol.

The false rejection probability, P_{FR}, and false acceptance probability, P_{FA}, in every authentication session due to the LPN problem can be defined as $P_{FR} = \sum_{i=\eta l+1}^{l} \binom{l}{i} \eta^i (1-\eta)^{l-i}$, and $P_{FA} = \sum_{i=0}^{\eta l} \binom{l}{i} 2^{-l}$.

4.1 Security

Theorem 1. Let $\text{LPN}_{\Pi}(\cdot)$ be a (t, ϵ_{Π})-secure LPN problem. The proposed PLFP protocol is (q, t_s, ϵ_s)-secure with $t_s = t - q\omega t_g - q(\omega+1)t_{\Pi}$ and $\epsilon_s = q\omega\left(\epsilon_g + \omega(2^{-l} + \epsilon_{\Pi})\right) + P_{FA}$.

Proof. The successful impersonation is the result of (i) adversary A can guess a valid instance of oracle $\Pi_{S,\eta}$ such that $A(c) = \text{LPN}_{\Pi}(S, \upsilon, c)$, and (ii) adversary A is falsely accepted.

We show that if there is an adversary A able to impersonate a legitimate tag with advantage at least ϵ_s within time at most t_s, then we can use adversary A as a subroutine to construct an algorithm B to distinguish $X(x) = \left(g(x), g^1(x), \ldots, g^{q\omega-1}(x)\right)$ from a purely random sequence of $q\omega n$-bit in time at most t_B with an advantage greater than ϵ_B.

One can easily check that when the input sequence received by B is $X(x)$, then B perfectly simulates the interaction between the tag, the reader, and the adversary A. Thus, the probability that B outputs '1' is equal to the advantage of A, ϵ_s. Let us analyze the probability that B outputs '1' while the input z_i is a truly random string. There are at most 1 valid pair for z_j, and at most q invalid pairs for z_j. Using Lemma 3, we could bound the possibility of guessing a valid $\Pi_{S,\eta}$ instance with the above information is upper-bounded by $q\omega(2^{-l} + \epsilon_{\Pi})$. The second reason is the coherent property of the LPN problem, which is upper-bounded by P_{FA}. Then we combine the two reasons to have $\epsilon_B \geq \epsilon_s - q\omega(2^{-l} + \epsilon_{\Pi}) - P_{FA}$. Using Theorem 1 in Berbain et al. (2009), we have $\epsilon_B \leq q\omega\epsilon_g$.

These two last inequalities show that $\epsilon_s \leq q\omega\left(\epsilon_g + \omega(2^{-l} + \epsilon_{\Pi})\right) + P_{FA}$.

4.2 Correctness

Theorem 2. Let $\text{LPN}_{\Pi}(\cdot)$ be a (t, ϵ_{Π})-secure LPN problem. The proposed PLFP authentication protocol is (q, t_c, ϵ_c)-correct with $t_c = t - q\omega t_g - q(\omega+1)t_{\Pi}$, $\epsilon_c = (L-1)\omega^2(2^{-l} + \epsilon_{\Pi} + P_{FA}) + L\omega\epsilon_g + L\epsilon_s + P_{FR}$.

Proof. There are two situations when a legitimate tag is rejected by the reader: (i) an unrecoverable desynchronization between them happens, and (ii) a false rejection happens.

Let us analyze the first situation. The first cause is a successful impersonation attack. In a system composed by L tags, the total possible impersonation advantage is upper-bounded by $L\epsilon_s$. The second cause is a collision event. According to Lemma 4, the collision probability of any two tag's responses is at most $(2^{-l} + \epsilon_{\Pi})$ in every session. Another reason of collision is the false acceptance, which is upper-bounded by P_{FA}. For each tag, the longest searching chain in every authentication is ω. So for a system with L tags with the longest search chain length $L\omega$, the collision probability of one undisturbed authentication of a legitimate tag is upper-bounded by $(L-1)\omega^2(2^{-l} + \epsilon_{\Pi} + P_{FA})$ in the case that the state sequence is uniformly random

strings. The second rejection situation is upper-bounded by P_{FR}.

At the end, combining the two rejection situations, the advantage of a correctness adversary can be represented as $\epsilon_c \leq (L-1)\omega^2(2^{-l}+\epsilon_{\Pi}+P_{FA})+L\omega\epsilon_g+L\epsilon_s+P_{FR}$.

4.3 Forward Privacy

Theorem 3. Let $\mathrm{LPN}_{\Pi}(\cdot)$ be a (t,ϵ_{Π})-secure LPN problem. The PLFP authentication protocol is (q,t_f,ϵ_f) -forward-private with $t_f = t - q\omega t_g - q(\omega+1)t_{\Pi}$ and $\epsilon_f = \frac{q-1}{2}(\epsilon_s + P_{FA}) + (3q+2)\epsilon_g$.

Proof. The following proof is composed of two steps. In the first step, we show adversary A can be used to construct a distinguisher B between a pseudo random number sequence and a real random number sequence, and in the second step we show that the two sequences cannot be distinguished with a non-negligible advantage.

First, the distinguisher B has an input of the as $X = (x_0,\dots,x_r,y_0,\dots,y_r,\dots y_{q-1},z_x,z_y)$. The X can take two different distributions, D and D'. In distribution D: the $(x_i)_{1\leq i\leq r}$, $(y_i)_{1\leq i\leq q-1}$, z_x and z_y are the pseudo random sequences constructed from two seeds s_x and s_y through $x_i \leftarrow g(x_{i-1})$ and $y_i \leftarrow g(y_{i-1})$. So the function of Distribution D is used to simulate the two tags $\mathcal{T}_0$ and $\mathcal{T}_1$. In distribution D': the sequences are random numbers.

For the distribution D, B correctly simulates the view of A except for successful impersonation attacks and false rejections. Let E denote the abnormal events in the simulation of B, and $\bar{E}$ denote the complementary event. The probability of E is $(q-r)(\epsilon_s+P_{FR})$. The probability that algorithm B outputs '1' with distribution D is analyzed as: $\Pr[B^X=1|X\in D] = \frac{1}{q}\sum_{r=1}^{q}\Pr[B^X=1|X\in D,r]$ and $\Pr[B^X=1|X\in D,r] \geq \Pr[B^X=1|X\in D,r,\bar{E}] - \Pr[E|X\in D,r]$, where $\Pr[B^X=1|X\in D,r,\bar{E}]$ represents the probability that algorithm B outputs '1' without abnormal events, so it equals the probability that adversary A successes, and $\Pr[E|X\in D,r]$ represents the probability of the abnormal events happening. So we have $\Pr[B^X=1|X\in D,r,\bar{E}] = \frac{1}{2}+\epsilon_f$, $\Pr[E|X\in D,r] = (q-r)(\epsilon_s+P_{FA})$, and $\Pr[B^X=1|X\in D] \geq \left(\frac{1}{2}+\epsilon_f\right) - \frac{q-1}{2}(\epsilon_s+P_{FA})$. In the case of distribution D', A is not able to distinguish between the two purely random sequences $(x_i)_{1\leq i\leq r}$ and $(y_i)_{1\leq i\leq q-1}$ and therefore $\Pr[B(X)=1|X\in D'] = 1/2$.

Combined with the above analysis, we obtain: $|\Pr[B^X=1|X\in D] - \Pr[B^X=1|X\in D']| \geq \epsilon_f - \frac{q-1}{2}(\epsilon_s+P_{FA})$. Therefore it shows that the probability of the distributions D and D' can be distinguished in time at most t_f+2qt_{Π} with an advantage at least $\epsilon_f - \frac{q-1}{2}(\epsilon_s+P_{FA})$. This can be written as: $\mathrm{Adv}_{D,D'}(t_f+2qt_{\Pi}) \geq \epsilon_f - \frac{q-1}{2}(\epsilon_s+P_{FA})$.

According Lemma 8 in Berbain et al. (2009), we use t_f+2qt_{Π} as input to obtain: $\mathrm{Adv}_{D,D'}(t_f+2qt_{\Pi}) \leq (3q+2)\epsilon_g$.

Combining the last two inequalities, we have $\epsilon_f \leq \frac{q-1}{2}(\epsilon_s+P_{FA})+(3q+2)\epsilon_g$.

5 ANALYSIS AND COMPARISON

Firstly, an analysis of the hardware cost of the tag is given. The secure PRNG can be implemented very efficiently thanks to the progress of stream ciphers in the eSTREAM project (Cid and Robshaw, 2009). To provide an 80-bit security level, Grain-v1 needs only 1,294 gates. Grain-v1 uses two shift register arrays for an initialization value (IV) and a state value as inputs, but the PRNG only needs one shift register array for its secret state as input. Therefore, if Grain-v1 is used, the hardware gate number will be less than 1,294 gates. The secret S and noise vector v do not need additional storage space, because they can be easily derived from the state and output of the PRNG. The inner product and XOR operation of the LPN problem can share the hardware with the PRNG, because Grain-v1 also uses XOR and AND operations. Then the overall crypto hardware cost is composed of a PRNG and some multiplexers. Therefore, the total estimated hardware cost should be no more than 2,000 gates, since the cost of multiplexers is no more than the cost of registers for the IV. Additionally, the hardware cost of the LPN problem is no more than a CRC function, so this protocol is suitable for EPC Class-1 Generation-2 tags (Juels, 2006), as the main hardware cost of such tags is a PRNG and a CRC.

Secondly, the proposed protocol is compared to the previous protocols, and the results are listed in Table 1 and 2. (i) It can be observed that this paper provides a high forward privacy level, that is to say, the forward privacy of all session can be protected. Although the O-FRAP (Le et al., 2007) and PEPS (Billet et al., 2010) protocols can provide infinite desynchronization resistance at the protocol level,

Table 1: Comparison Results 1.

	Billet et al., 2010	Cao, O'Neill, 2011	Le et al., 2007
Forward Privacy level	limited	limited	limited
Proof model	standard	standard	UC
Desynchronization Resistance	Infinite	Infinite	Infinite
Crypto hardware cost	≈ 2,000 gates	≈ 3,500 gates	≈ 4,000 gates

Table 2: Comparison Results 2.

	Berbain et al., 2009	Avoine, Oechslin, 2005	This paper
Forward Privacy level	high	high	high
Proof model	standard	random oracle	standard
Desynchronization Resistance	$< \omega$	$< \omega$	$< \omega$
Crypto hardware cost	≈ 3,500 gates	> 5,000 gates	≤ 2,000 gates

the trade-off is that they provide a limited forward privacy (i.e., only the forward privacy of successful sessions is assured). (ii) The hardware cost of the proposed protocol is similar to that of the PEPS protocol. But the proposed protocol provides a higher forward privacy level. (iii) The F-HB (Cao and O'Neill, 2011) uses the PRNG to implement a message authentication code, so it consumes more hardware than the proposed protocol.

6 CONCLUSIONS

In this paper, the PRNG and LPN problem are proposed to construct a provable secure and forward private authentication protocol. The PRNG and LPN problem can be efficiently implemented on a tag using only one low-cost stream cipher, so that this protocol is suitable for low-cost RFID applications. According to the hardware analysis, the proposed protocol can meet the hardware constraints of the EPC Class-1 Generation-2 tags, provide a high forward privacy, and require a small hardware cost on the tag.

RFERENCES

Avoine, G. and Oechslin, P. 2005. A Scalable and Provably Secure Hash-Based RFID Protocol. *IEEE International Workshop on Pervasive Computing and Communication Security*.

Berbain, C., Billet, O., Etrog, J. and Gilbert, H. 2009. An Efficient Forward Private RFID Protocol. *ACM Conference on Computer and Communications Security (CCS)*.

Billet, O., Etrog, J. and Gilbert, H. 2010. Lightweight Privacy Preserving Authentication for RFID Using a Stream Cipher. *International Workshop on Fast Software Encryption (FSE)*

Cid, C. and Robshaw, M. 2009. *The eSTREAM Portfolio 2009 Annual Update*. Available from: http://www.ecrypt.eu.org/stream/.

Cao, X. and O'Neill, M. 2011. F-HB: An Efficient Forward Private Protocol. *Workshop on Lightweight Security & Privacy: Devices, Protocols and Applications (Lightsec2011)*.

Gilbert, H., Robshaw, M. J. B. and Seurin, Y. 2008. HB#: Increasing the Security and Efficiency of HB+. *Annual International Conference on the Theory and Applications of Cryptographic Techniques, EUROCRYPT*.

Hopper, N. J. and Blum, M. 2001. Secure Human Identification Protocols. *International Conference on the Theory and Application of Cryptology and Information Security*, *ASIACRYPT*.

Juels, A. and Weis, S. A. 2005. Authenticating Pervasive Devices with Human Protocols. *International Cryptology Conference, CRYPTO*.

Juels, A. 2006. RFID Security and Privacy: A research Survey. *IEEE Journal on Selected Areas in Communications*.

Katz, J. and Shin, J. S. 2006. Parallel and Concurrent Security of the HB and HB+ Protocols. *Annual International Conference on the Theory and Applications of Cryptographic Techniques, EUROCRYPT*.

Le, T. V., Burmester, M. and de Medeiros, B. 2007. Universally Composable and Forward-secure RFID Authentication and Authenticated Key Exchange. *ACM Symposium on InformAtion, Computer and Communications Security (ASIACCS)*.

Ohkubo, M., Suzuki, K. and Kinoshita, S. 2003. Cryptographic Approach to Privacy-Friendly Tags. *RFID Privacy Workshop*.