Security Analysis of the SASI Protocol

Tianjie Cao, Elisa Bertino, *Fellow, IEEE*, and Hong Lei

Abstract—The ultralightweight RFID protocols only involve simple bit-wise operations (like XOR, AND, OR, etc.) on tags. In this paper, we show that the ultralightweight strong authentication and strong integrity (SASI) protocol has two security vulnerabilities, namely denial-of-service and anonymity tracing based on a compromised tag. The former permanently disables the authentication capability of a RFID tag by destroying synchronization between the tag and the RFID reader. The latter links a compromised tag with past actions performed on this tag.

Index Terms—Authentication, location-dependent and sensitive, security and privacy protection

1 Introduction

Radio Frequency Identification (RFID) systems are a common and useful tool for admission control, payment, ticketing and supply chain management. However, several security and privacy concerns have been identified in connection with the use of RFIDs.

A RFID system typically consists of two components: a set of tags, also called transponders, and a set of readers, also called transceivers. Tags are attached to physical objects. Readers query these tags for some (potentially unique) identifying information about the objects to which tags are attached. Although readers are often regarded as a simple conduit to a back-end database, for simplicity we treat a reader and a back-end database as a single entity. A key security problem in such a context is that an adversary A can arbitrarily modify the conversations between any pair of tag and reader, and indeed initiates and terminates a session at its choice.

The security of a RFID protocol can be described in terms of four games, an authentication game G_{auth}, an anonymity game G_{anon}, a forward anonymity game G_{fanon} and an availability game G_{avail}, with the following players: the malicious adversary A against the honest tags and the honest readers. These games have two steps. The first step is a preparing step for adversary A: A is allowed to interact arbitrarily with the tags and the readers. In the second step, A's knowledge is tested. The score of A in game G is its advantage adv_G^A. A wins if its advantage is non-negligible. We now describe in more detail the second steps of the four games: G_{auth}, G_{anon}, G_{fanon} and G_{avail}.

Authentication: In the second step of G_{auth}, A must impersonate some tag T to some reader R. During this impersonation step, A is allowed to interact arbitrarily with all other tags and readers, except tag T that A is trying to impersonate. The advantage adv_{Gauth}^A of the adversary is the probability that A succeeds in authenticating itself to R. An RFID protocol is a secure authentication protocol if adv_{Gauth}^A is negligible. Impersonation is an attack on authentication.

Anonymity: Anonymity means that given two interactions A is not able to say whether they are with the same tag T. For anonymity we require that the advantage adv_{Ganon}^A of the adversary in the second step of G_{anon} in linking two different interactions with the same tag is negligible. Anonymity property is also called untraceability, unlinkability, or indistinguishability.

Forward anonymity: Forward anonymity means that even if the adversary obtains the secret data stored at a tag by tampering with the tag, the adversary's advantage adv_{Gfanon}^A in the second step of G_{fanon} in tracing the data back is negligible. The authentication transcripts of the tag should not be traced back using previous known messages, i.e., disclosed data and communication information. Forward anonymity is often called forward security or forward untraceability. Tracing is an attack on forward anonymity.

Availability: In G_{avail} the adversary A must prevent a tag T from being authenticated by a reader R in a challenge session *ses*, without interacting with the session *ses*. In this attack, A is allowed to interact with all tags and readers, except of course for the session *ses*. The advantage adv_{Gavail}^A of A in this game is the probability that R rejects T in the challenge session *ses*. An RFID protocol is an availablility-assuring protocol if adv_{Gavail}^A is negligible. Denial of service (DoS) is an attack against the availability. Especially, the de-synchronization attack carried out by a man-in-the-middle attack must be prevented.

To deal with the above security threats, many authentication protocols for RFID tags have been proposed so far. RFID tags are generally low cost with extremely limited resources, so they cannot perform the public key algorithms. Most previous protocols require the support of either hash function or symmetric encryption on the tag. The lightweight RFID authentication protocols require a random number generator and simple functions like cyclic redundancy code (CRC) checksum [1], [2], [3], [4], [5]. Some weaknesses of these schemes have been recently reported [6], [7], [8]. The ultralightweight protocols only involve simple bit-wise operations (like

- *T.J. Cao and H. Lei are with the School of Computer, Nanhu Campus, China University of Mining and Technology, Xuzhou, 221116, China. E-mail: tjcao@cumt.edu.cn, Faith_Lei@satyam.com.*
- *E. Bertino is with the Center for Education and Research in Information Assurance and Security (CERIAS) and also with the Department of Computer Sciences, Purdue University, West Lafayette, IN 47907-2107. E-mail: bertino@cs.purdue.edu.*

Manuscript received (insert date of submission if desired). Please note that all acknowledgments should be placed at the end of the paper, before the bibliography.

involve simple bit-wise operations (like XOR, AND, OR, etc.) on tags [9], [10], [11], [12], [13], [14]. However, de-synchronization attack and the full-disclosure attack against such protocols have been reported [9], [10], [11].

Recently, Chien proposed the ultralightweight strong authentication and strong integrity (SASI) protocol where the tag requires only simple bit-wise operations [15]. Since the tag does not support random number generator to generate a challenge nonce, an attacker can replay old messages and impersonates a reader. Thus, the assertion of the SASI protocol that it provides mutual authentication is incorrect. In [15], Chien claimed that the SASI protocol is resistant to the de-synchronization attack and man-in-the-middle attack. However, we show that the SASI protocol is prone to DoS attacks. In our attacks, a man-in-the-middle can destroy the synchronization between the database and the tag. Thus, the tag cannot be further authenticated by the database. The RFID system will be involved in DoS state and unable to guarantee availability. Chien also claimed that the SASI protocol satisfies forward security. However, if we assume that an attacker compromises a tag, the attacker can infer the previous secret data and keys of the same tag and trace the past communication. Thus, the SASI protocol does not provide forward anonymity.

The rest of this paper is organized as follows. We review the SASI protocol in Section 2 and analyze two vulnerabilities in Section 3. In Section 4, we conclude the paper.

2 Review of the SASI Protocol

In the SASI protocol [15], each tag has a static identifier (*ID*), and preshares a pseudonym (*IDS*) and two keys $K1$, $K2$ with the backend database. The length of each of *ID*, *IDS*, $K1$, $K2$ is n bits. Typically, the value n is 96. Each tag keeps two entries of the form (*IDS*, $K1$, $K2$): one is for the old values for the pseudonym and two keys, and the other is for the potential next values. SASI is a highly efficient RFID authentication protocol using only bitwise XOR ($\oplus$), bitwise OR ($\vee$), bitwise AND ($\wedge$), addition mod 2^n ($+$) and left rotate ($Rot(x, y)$) operations. $Rot(x, y)$ left rotates the value of x with y bits. Expensive operations, such as multiplications and hash functions, are not required at all by SASI, and random number generation is only executed by the reader. SASI assumes that the channel between the reader and the backend database is secure, but that the channel between the reader and the tag is susceptible to all the possible attacks. The specification of the SASI protocol is shown in Fig. 1.

The protocol has three phases: tag identification, mutual authentication, pseudonym update and key update.

Tag identification: Initially, the reader sends "*hello*" to the tag, which then responds with its potential next *IDS*. If the reader can find a matched entry in the database, it starts the mutual authentication phase; otherwise, it probes the tag again and the tag responds with its old *IDS*.

Mutual authentication: The reader uses the matched values and two randomly generated integers $n1$ and $n2$ to compute the values A, B and C (the calculation equations are specified in Fig. 1). Such values are then sent to the tag. From $A||B||C$, the tag first extracts $n1$ from A, extracts $n2$ from B, computes $\overline{K1}$ and $\overline{K2}$, and then computes the response value D. Upon receiving D, the reader uses its local values to verify D.

Pseudonym update and key update: After the reader and the tag authenticate each other, they update their local pseudonym and keys.

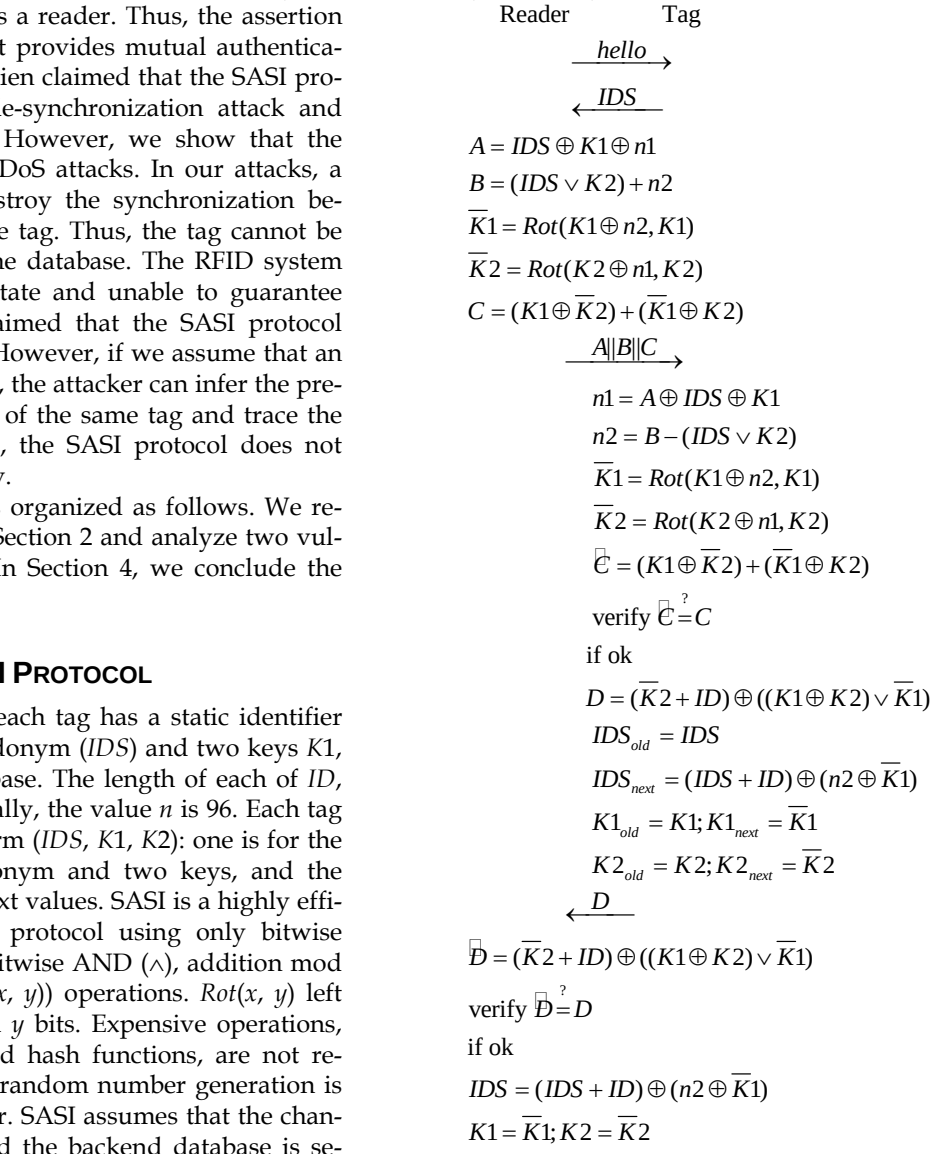

Fig. 1. The SASI Protocol

3 Vulnerabilities of the SASI Protocol

We assume that there is a completion message exchanged between the tag and the reader to indicate a successful completion of the protocol. This completion message will enable the update operations at both the reader and the tag.

3.1 DoS Attack

In general, a DoS attack results in loss of service to users. In other words, the attacker does not try to obtain infor-

mation, but rather it tries to prevent a legitimate reader from accessing data stored in tags. To assure untraceability for a RFID tag, the SASI protocol updates the database's secret information, that is, IDS, $K1$ and $K2$ after a successful protocol run. The tag updates the secret information accordingly so that a reader can still authenticate the tag later on. So the synchronization of secret information between the database and the tag is crucial to resist to DoS attacks.

In the SASI protocol, if the current secret information $K1_{next}$, $K2_{next}$, $K1_{old}$, $K2_{old}$ for a tag is different from the key $K1$ and $K2$ stored in the database, the tag will be in a *de-synchronization state* with respect to the database leading to a DoS situation. In what follows, we show attacks that lead to a de-synchronization state for the tag.

Attack 1: Changing messages A, C and D. An attacker can first eavesdrop on the on-going protocol, and then replace $A||B||C$ with $A'||B||C'$, where $A'=A\oplus[I]_0$, $C'=C\oplus[I]_0$ and $[I]_0$=[000…001] (set the first n-1 most significant bits of I as 0 and the least significant bit as 1). Similarly, the attacker changes the reply D from the tag to $D'=D\oplus[I]_0$. This procedure is specified in Table 1.

TABLE 1
CHANGING MESSAGES A AND C

Reader→Tag: *hello* Tag→Reader: *IDS* Reader→Tag(Attacker): $A||B||C$ Reader(Attacker)→Tag: $A'||B||C'$ Tag→Reader(Attacker): D Tag(Attacker)→Reader: D'
where: $A'=A\oplus[I]_0$ $C'=C\oplus[I]_0$ $D'=D\oplus[I]_0$

We now analyze the success rate of such an attack:

(1) Once the tag receives $A'||B||C'$, the probability that the tag accepts the message $A'||B||C'$ is not less than $1/(2n)$.

Suppose that $K2$ is a random number; there is a probability equal to $1/n$ that $K2$ mod n = 0 and a 1/2 probability that the least significant bit of $K2\oplus\overline{K1}$ be 0. We note that for any X there is $Rot(K2\oplus X, K2) = K2\oplus X$ when $K2$ mod n =0. In this case, we check the validity of the message $A'||B||C'$.

$$\begin{aligned}C' &= C\oplus[I]_0 \\ &= [(K1\oplus\underline{K}2)+(\overline{K}1\oplus\underline{K}2)]\oplus[I]_0 \\ &= (K1\oplus\overline{K}2\oplus[I]_0)+(\overline{K}1\oplus K2)\end{aligned}$$

$$\begin{aligned}n1' &= A'\oplus IDS\oplus K1 \\ &= A\oplus[I]_0\oplus IDS\oplus K1 \\ &= (IDS\oplus K1\oplus n1)\oplus[I]_0\oplus IDS\oplus K1 \\ &= n1\oplus[I]_0\end{aligned}$$

The operation on A is actually toggling the least significant bit of $n1$.

$$\begin{aligned}n2' &= B-(IDS\vee K2) \\ &= (IDS\vee K2)+n2-(IDS\vee K2) \\ &= n2\end{aligned}$$

$$\overline{K}1' = Rot(K1\oplus n2, K1) = \overline{K}1$$

$$\overline{K}2' = Rot(K2\oplus n1', K2) = K2\oplus n1' = \overline{K}2\oplus[I]_0$$

$$\begin{aligned}C' &= (K1\oplus\underline{\overline{K}2'})+(\underline{\overline{K}1'}\oplus K2) \\ &= (K1\oplus\overline{K}2\oplus[I]_0)+(\overline{K}1\oplus K2) \\ &= C'\end{aligned}$$

In the case in which $K2$ mod n =0 and the least significant bit of $K2\oplus\overline{K}1$ is 0, the tag will accept the message $A'||B||C'$.

(2) Once the reader receives D', the probability that the reader accepts the message D' is not less than 1/2.

If the least significant bit of ID is 0, the reader will accept D'. There is a 1/2 probability that the least significant bit of ID is 0. We have

$$\begin{aligned}D' &= \underline{D}\oplus[I]_0 \\ &= (\underline{\overline{K}2'}+ID)\oplus((K1\oplus K2)\vee\overline{K}1)\underline{\oplus}[I]_0 \\ &= (\underline{\overline{K}2}\oplus[I]_0+ID)\oplus((K1\oplus K2)\vee K1)\oplus\underline{[I]}_0 \\ &= ((\underline{\overline{K}2}\oplus[I]_0+ID)\oplus[I]_0)\oplus(\underline{(K1}\oplus K2)\vee K1) \\ &= (\underline{\overline{K}2}+ID)\oplus((K1\oplus K2)\vee K1) \\ &= D\end{aligned}$$

Once the reader accepts the value, the reader needs to update the tag's secret information with the pair ($n1$, $n2$). However, the tag uses another pair ($n1\oplus[I]_0$, $n2$) to update its secrets. It is obvious that there is a mismatch between the secrets stored at the tag and at the reader. So there is a non-negligible probability value, that is, $(1/n)*(1/2)*(1/2)=1/(4n)$ in succeeding in a DoS attack. In fact, this attack can be extended to toggle a single bit of A at any location i, so that it can be a general attack with the same $1/(4n)$ success probability.

Attack 2: Changing messages B and C. The attacker can first eavesdrop on the on-going protocol, and then replace $A||B||C$ with $A||B'||C'$, where $B'=B+1$, $C'=C\oplus[I]_0$. This procedure is specified in Table 2.

At the tag side, the attack does not affect the first round of the interaction protocol, that is, "tag identification". But in the second round, when the tag receives the message $A||B'||C'$, it can still authenticate the reader with a non-negligible probability. But, the tag will receive a wrong random number $n2'$ (where $n2'$ depends on $n2$). The tag will accept this value and compute its reply according to $n2'$. In this attack, the attacker can now provide the reader with a reply D. If the reader accepts value D, the attack is successful; otherwise, the attack fails. Now we analyze the success rate:

TABLE 2
CHANGING MESSAGES B AND C

Reader→Tag: *hello* Tag→Reader: *IDS* Reader→Tag(Attacker): $A||B||C$ Reader(Attacker)→Tag: $A||B'||C'$ Tag→Reader: D
where: $B'=B+1$ $C'=C\oplus[I]_0$

(1) Once the tag receives $A||B'||C'$, the probability that the tag accepts the message $A||B'||C'$ is not less than $1/(4n)$.

Suppose that $K1$ is a random number, there is a probability equal to $1/n$ that $K1$ mod n =0 and a 1/4 probability that the least significant bit of $K1\oplus\underline{K}2$ and $n2$ are

0 simultaneously. We note that for any X there is $Rot(K1\oplus X, K1) = K1\oplus X$ when $K1 \bmod n = 0$. In this case, we check the validity of the massage $A||B'||C'$.

$C' = C\oplus[I]_0$
$= [(K1\oplus \underline{K}2)+(\overline{K}1\oplus K2)]\oplus[I]_0$
$= (K1\oplus \underline{K}2)+(K1\oplus K2\oplus[I]_0)$

$n1' = A \oplus IDS\oplus K1$
$= A\oplus IDS\oplus K1$
$= (IDS\oplus K1\oplus n1) \oplus IDS\oplus K1$
$= n1$

$n2' = B'-(IDS\vee K2)$
$= B+1-(IDS\vee K2)$
$= ((IDS\vee K2)+n2)+1-(IDS\vee K2)$
$= ((IDS\vee K2)+(n2\oplus[I]_0))-(IDS\vee K2)$
$= n2\oplus[I]_0$

The operation on B is actually toggling the least significant bit of $n2$.

$\underline{K}1' = Rot(K1\oplus n2', K1) = \underline{K}1\oplus n2' = \overline{K}1\oplus[I]_0$
$\underline{K}2' = Rot(\underline{K}2\oplus n1', \underline{K}2) = \underline{K}2$
$C' = (K1\oplus \underline{K}2')+(\underline{K}1'\oplus K2)$
$= (K1\oplus \underline{K}2)+(\overline{K}1\oplus[I]_0\oplus K2)$
$= C'$

In the case in which $K1 \bmod n = 0$ and the least significant bit of $K1\oplus \underline{K}2$ and $n2$ is 0, the tag will accept the message $A||B'||C'$.

(2) Once the reader receives D, the probability that the reader accepts the message D is not less than 1/2.

If the least significant bit of $(K1\oplus K2)$ is 1, the reader will accept D. There is a 1/2 probability that the least significant bit of $(K1\oplus K2)$ is 1. We have

$D = (\underline{K}2'+ID)\oplus((K1\oplus K2)\vee\underline{K}1')$
$= (\underline{K}2+ID)\oplus((K1\oplus K2)\vee(\underline{K}1\oplus[I]_0)$
$= (\underline{K}2+ID)\oplus((K1\oplus K2)\vee K1)$
$= D'$

Once the reader accepts the value, the reader needs to update the tag's secret information with the pair $(n1, n2)$. However, the tag uses another pair $(n1, n2\oplus[I]_0)$ to update its secrets. It is obvious that there is a mismatch between the secrets stored at the tag and at the reader. So there is a non-negligible probability value, that is, $(1/n)*(1/4)*(1/2)=1/(8n)$, that the attacker succeeds in a DoS attack.

Attack 3: Changing A and guessing C. The above attacks are basically man-in-the-middle attacks on the communication between the tag and the reader. We now introduce another attack in which the attacker pretends to be a valid reader by transmitting messages to the tag using an eavesdropped message. If the tag authenticates the reader, that is, the attacker, and updates its values, the DoS will succeed.

The first step: The reader sends "*hello*" to the tag, which responds with its potential *IDS*. Then the attacker records message $A||B||C$. After the authentication and secrets update, the reader will hold the new values $K1$ and $K2$, and the tag will hold the old values $K1_{old}$, $K2_{old}$ and the new values $K1_{next}$, $K2_{next}$.

The second step: Let $[I]_0=[000\ldots001]$, $[I]_1=[000\ldots010]$, …, $[I]_{n-1}=[100\ldots000]$. The attacker changes A to A' where $A'=A\oplus[I]_0$. In this case, the attacker guesses all possible values of $C_{0i}=C+[I]_i$ and $C_{1i}=C-[I]_i$ $(i = 0,1,\ldots, n-1)$. The details of the attack are given in Table 3.

TABLE 3
CHANGING *A* AND GUESSING *C*

```
for i = 0 to n-1
  for j = 0 to 1
   {sends hello to the tag;
   receives IDS_next from the tag;
   sends a random number to the tag;
   sends hello to the tag;
   receives IDS_old from the tag;
   sends A'||B||C_ji to the tag;
   if receives D from the tag,  returns success
   }
```

The attacker sends "*hello*" to the tag, which first responds with its IDS_{next}. The attacker then sends a random number to the tag. This mutual authentication fails. Next time, the attacker sends "*hello*" to the tag, which responds with the tag's old value IDS_{old}. The attacker sends the guessed value $A'||B||C_{ji}$ to the tag and observes the replies from the tag. If the tag sends a message D, it means that the attack is successful; if not, the attack continues.

Let $i = K2 \bmod n$, we prove that the tag will accept $A'||B||C_{0i}$ or $A'||B||C_{1i}$ where $A'=A\oplus[I]_0$, $C_{0i}=C+[I]_i$ and $C_{1i}=C-[I]_i$.

Similar to the analysis in attack 1, we have:

$n1' = n1\oplus[I]_0$
$n2' = n2$
$\underline{K}1' = Rot(K1\oplus n2, K1) = \overline{K}1$
$\underline{K}2' = Rot(K2\oplus n1', K2)$
$= Rot(K2\oplus n1\oplus[I]_0, i)$
$= \underline{Rot}(K2\oplus n1, i) \oplus Rot([I]_0, i)$
$= \underline{K}2\oplus[I]_i$
$C' = (K1\oplus \underline{K}2')+(\overline{K}1'\oplus K2)$
$= (K1\oplus \underline{K}2\oplus[I]_i)+(\overline{K}1\oplus K2)$
$C_{0i} = C+[I]_i$
$= [(K1\oplus \underline{K}2)+(\overline{K}1\oplus K2)]+[I]_i$
$C_{1i} = C+[I]_i$
$= [(K1\oplus \underline{K}2)+(\overline{K}1\oplus K2)]-[I]_i$

If the i-th least significant bit of $K1\oplus \overline{K}2$ is 0, then C' is equal to C_{0i} else C' is equal to C_{1i}.

Once the tag accepts the guessed value, the tag will update the tag's secret information with the pair $(n1\oplus[I]_0, n2)$. However, the reader has updated the secrets with another pair $(n1, n2)$. It is obvious that there is a mismatch between the secrets stored at the tag and at the reader. There are at most $2n$ guesses to succeed in such DoS attack.

3.2 Tracing Attack

RFID tags are inexpensive devices that offer no tamper resistance. Hence an attacker upon compromising a tag may be able to read its secret values and link this tag with past actions performed on the tag. With forward anonymity, disclosure of current secret key material does not compromise the secrecy of earlier material. The SASI protocol does not provide forward anonymity.

A communication view of the protocol is defined to be the set of all messages that the reader has received and

generated when authenticating a tag. The attacker can construct the communication view database by recording all the authentication transcripts between the reader and the tags. For each instance i of the protocol, the attacker can record (IDS_i, A_i, B_i, C_i, D_i) when the tag communicates with the reader during the instance i of the protocol. The tuple (IDS_i, A_i, B_i, C_i, D_i) is referred to as the view by the attacker on the instance i of the protocol. We suppose that the communication view database has N records.

Suppose that a tag is compromised through a physical attack after the authentication phase. Then the attacker would get the values ID, IDS_m, $K1_m$, $K2_m$, IDS_{m+1}, $K1_{m+1}$ and $K2_{m+1}$ of the tag. To link the values (ID, IDS_{m+1}, $K1_{m+1}$, $K2_{m+1}$) and the past communication, the attacker can easily trace the last authentication view by searching the communication view database using the condition $IDS = IDS_m$. Now we introduce an algorithm to find the tuple record in the communication view database that links to (ID, IDS_m, $K1_m$, $K2_m$).

TABLE 4
TRACING ATTACK

```
for i=0 to N-1
{get the i-th record (IDS_i, A_i, B_i, C_i, D_i) from view
database
n2=(IDS_m⊕(IDS_i+ID))⊕K1_m;
for j=0 to n-1
   {K1_{m-1}= Rot(K1_m, j)⊕n2;
    if (K1_m=Rot(K1_{m-1}⊕n2, K1_{m-1}))
       {K2_{m-1}=(C_i - ( K1_{m-1}⊕K2_m))⊕K1_m;
        n1= A_i⊕IDS_i⊕K1_{m-1};
        B'=(IDS_i∨ K2_{m-1})+n2;
        D'=(K2_m+ID)⊕(( K1_{m-1}⊕ K2_{m-1})∨K1_m)
        if(B'=B_i &&D'=D_i)
        return (i, IDS_i, K1_{m-1}, K2_{m-1});
        }
    }
}
```

For each view record (IDS_i, A_i, B_i, C_i, D_i), the attacker checks whether or not it links to (ID, IDS_m, $K1_m$, $K2_m$). The attacker computes $(IDS_m \oplus (IDS_i + ID)) \oplus K1_m$ to derive the value $n2$. Once the attack obtains $n2$, it can compute $K1_{m-1}$ from equation $K1_m = Rot(K1_{m-1} \oplus n2,\ K1_{m-1})$. The attacker computes all potential candidates $K1_{m-1} = Rot(K1_m,\ j) \oplus n2$ ($0 \le j \le n-1$), and checks whether $K1_m$ equals to $Rot(K1_{m-1} \oplus n2,\ K1_{m-1})$. If such two values match, the attacker computes $K2_{m-1} = (C_i - (K1_{m-1} \oplus K2_m)) \oplus K1_m$ and $n1 = A_i \oplus IDS_i \oplus K1_{m-1}$, and checks whether B_i equals to $(IDS_i \vee K2_{m-1}) + n2$ and D_i equals to $(K2_m + ID) \oplus ((K1_{m-1} \oplus K2_{m-1}) \vee K1_m)$. If the above equations hold, the attacker succeeds. Otherwise the attacker checks the next view record. Once the attacker obtains (i, IDS_i, $K1_{m-1}$, $K2_{m-1}$), it can use the same algorithm to trace the former communication view. Therefore, the SASI protocol does not provide forward anonymity; the past communication from the same tag can be traced.

4 CONCLUSIONS

In this paper, we have demonstrated two effective attacks against the SASI protocol recently proposed in [15]. The severity of the attacks indicates the insecure design of the protocol. Our work shows that it may be quite dangerous using only simple bitwise operations to achieve RFID authentication under powerful adversarial model. The security of such protocols must be proved with careful cryptanalysis. How to design a secure protocol without strong cryptographic algorithms such as hash function and symmetric encryption is an open problem. We plan, as our next step, to design a secure (ultra) lightweight RFID mutual authentication protocol that keeps these attacks into account, and to apply it to low-cost RFID tags.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their helpful comments. This work is partially supported by the US National Science Foundation under grant 0712846 "IPS: Security Services for Healthcare Applications", the Jiangsu Provincial Natural Science Foundation of China (BK2007035) and the Science and Technology Foundation of CUMT.

REFERENCES

[1] A. Juels and S. A. Weis, "Authentication pervasive device with human protocols", In *Advance in Cryptology – CRYPTO2005*. Springer-Verlag. Lecture Notes in Computer Science, Vol. 3621, pp. 293-308, 2005

[2] S. A. Weis, "Security parallels between people and pervasive devices", In *PERCOMW'05: Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications Workshop*, Washington, DC, USA. IEEE Computer Society, pp. 105-109, 2005, doi:10.1109/PERCOMW.2005.72.

[3] H. Briger, H. Chabanne, and E. Dottax, "HB++: A Lightweight Authentication Protocol Secure against Some Attacks," In *Proc. IEEE Int'l Conf. Pervasive Service, Workshop Security, Privacy and Trust in Pervasive and Ubiquitous Computing*, pp. 28-33, 2006, doi:10.1109/SECPERU.2006.10.

[4] D.N. Duc, J. Park, H. Lee, and K. Kim, "Enhancing Security of EPCglobal Gen-2 RFID Tag against Traceability and Cloning," In *Proc. 2006 Symp. Cryptography and Information Security*, 2006

[5] A. Juels, "Strengthening EPC Tag against Cloning," In *Proc. ACM Workshop Wireless Security (WiSe '05)*, pp. 67-76, 2005.

[6] H. Gibert, M. Robshaw, and H. Sibert, "An active attack against HB+ - a provably secure lightweight authentication protocol", *IEE Electronics Letters*, vol. 41, no. 21, pp.1169-1170, October 2005, doi:10.1049/el:20052622.

[7] H.-Y. Chien and C.-H. Chen, "Mutual Authentication Protocol for RFID Conforming to EPC Class 1 Generation 2 Standards," *Computers Standards & Interfaces*, vol. 29, no. 2, pp 254-259, 2007, doi:10.1016/j.csi.2006.04.004.

[8] S. Piramuthu, "Protocols for RFID tag/reader authentication", *Decision Support Systems*, vol. 43, no. 3, pp. 897-914, 2007, doi:10.1016/j.dss.2007.01.003.

[9] H.-Y. Chien and C.-W. Huang, "Security of Ultra-Lightweight RFID Authentication Protocols and Its Improvements," *ACM Operating System Rev.*, vol. 41, no. 2, pp. 83-86, July 2007, doi: http://doi.acm.org/10.1145/1278901.1278916.

[10] T. Li and R.H. Deng, "Vulnerability Analysis of EMAP-An Efficient RFID Mutual Authentication Protocol," In *Proc. Second Int'l Conf. Availability, Reliability, and Security (AReS '07)*, 2007.

[11] T. Li and G. Wang, "Security Analysis of Two Ultra-Lightweight RFID Authentication Protocols," In *Proc. 22nd IFIP TC-11 Int'l Information Security Conf.,* May 2007.
[12] P. Peris-Lopez, J.C. Hernandez-Castro, J.M. Estevez-Tapiador, and A. Ribagorda, "LMAP: A Real Lightweight Mutual Authentication Protocol for Low-Cost RFID Tags," In *Proc. Second Workshop RFID Security*, July 2006.
[13] P. Peris-Lopez, J.C. Hernandez-Castro, J.M. Estevez-Tapiador, and A. Ribagorda, "EMAP: An Efficient Mutual Authentication Protocol for Low-Cost RFID Tags," In *Proc. OTM Federated Conf. and Workshop: IS Workshop*, Nov. 2006.
[14] P. Peris-Lopez, J.C. Hernandez-Castro, J.M. Estevez-Tapiador, and A. Ribagorda, "M^2AP: A Minimalist Mutual-Authentication Protocol for Low-Cost RFID Tags," In *Proc. Int'l Conf. Ubiquitous Intelligence and Computing (UIC'06)*, pp. 912-923 2006.
[15] H.-Y. 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, 2007, doi:10.1109/TDSC.2007.70226.

Tianjie Cao received the BS and MS degree in mathematics from Nankai University, Tianjin, China and the PhD degree in computer software and theory from State Key Laboratory of Information Security of Institute of Software, Chinese Academy of Sciences, Beijing, China. He is a professor of computer science in the School of Computer Science and Technology, China University of Mining and Technology, Xuzhou, China. From 2007 to 2008, he has been a visiting scholar at the Department of Computer Sciences and CERIAS, Purdue University. His research interests are in security protocols and network security.

Elisa Bertino is a professor of computer science in the Department of Computer Sciences, Purdue University and the Research Director of the Center for Education and Research in Information Assurance and Security (CERIAS). Previously, she was a faculty member in the Department of Computer Science and Communication, University of Milan, where she directed the DB and SEC Laboratory. She was a visiting researcher at the IBM Research Laboratory (now Almaden), San Jose, at the Microelectronics and Computer Technology Corporation, at Rutgers University, and at Telcordia Technologies. From 2001 to 2007, she was a coeditor in chief of the Very Large Database Systems (VLDB) Journal. She serves also on the editorial boards of several scientific journals, including the IEEE Internet Computing, IEEE Security and Privacy, ACM Transactions on Information and System Security, and ACM Transactions on Web. Her main research interests include security, privacy, digital identity management systems, database systems, distributed systems, multimedia systems. She has published more than 250 papers in all major refereed journals and in the proceedings of international conferences and symposia. She is a coauthor of Object-Oriented Database Systems: Concepts and Architectures (Addison-Wesley, 1993), Indexing Techniques for Advanced Database Systems (Kluwer Academic Publishers, 1997), Intelligent Database Systems (Addison-Wesley, 2001), and Security for Web Services and Service Oriented Architectures (Springer, Fall 2007). She is a fellow of the IEEE and the ACM and a Golden Core member of the IEEE Computer Society. She received the 2002 IEEE Computer Society Technical Achievement Award for her "outstanding contributions to database systems and database security and advanced data management systems" and the 2005 IEEE Computer Society Tsutomu Kanai Award for "pioneering and innovative research contributions to secure distributed systems."

Hong Lei is currently working toward the Master degree in the School of Computer Science and Technology, China University of Mining and Technology.