Hindawi
Wireless Communications and Mobile Computing
Volume 2019, Article ID 2138468, 8 pages
https://doi.org/10.1155/2019/2138468

Research Article

Inductive Method for Evaluating RFID Security Protocols

Defu Liu,[1] Guowu Yang,[1,2] Yong Huang ,[2] and Jinzhao Wu[2,3]

[1]*Big Data Research Center, School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China*
[2]*Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis, Guangxi University for Nationalities, Nanning, 530006, China*
[3]*Guangxi University, Nanning, 530004, China*

Correspondence should be addressed to Yong Huang; gxunhy@163.com

Received 15 November 2018; Accepted 21 March 2019; Published 11 April 2019

Academic Editor: Antonio Guerrieri

Authentication protocol verification is a difficult problem. The problem of "state space explosion" has always been inevitable in the field of verification. Using inductive characteristics, we combine mathematical induction and model detection technology to solve the problem of "state space explosion" in verifying the OSK protocol and VOSK protocol of RFID system. In this paper, the security and privacy of protocols in RFID systems are studied and analysed to verify the effectiveness of the combination of mathematical induction and model detection. We design a (r,s,t)-security experiment on the basis of privacy experiments in the RFID system according to the IND-CPA security standard in cryptography, using mathematical induction to validate the OSK protocol and VOSK protocol. Finally, the following conclusions are presented. The OSK protocol cannot resist denial of service attacks or replay attacks. The VOSK protocol cannot resist denial of service attacks but can resist replay attacks. When there is no limit on communication, the OSK protocol and VOSK protocol possess (r,s,t)-privacy; that is to say they can resist denial of service attacks.

1. Introduction

Radio frequency identification (RFID) is the wireless use of electromagnetic fields to transfer data to automatically identify and track tags attached to objects. Compared with traditional bar codes, the RFID system has many advantages, such as noncontact scanning and long lifespan. RFID also has many applications, such as animal breeding management, access control, retail, and manufacturing management [1–4]. RFID can be used to enhance visibility and traceability in supply chains. Once a product is affixed with an RFID tag at the beginning of the supply chain, it is assigned a unique electronic product code (EPC) and can be automatically identified, tracked, and traced from the supplier to the customer [5]. RFID technology is the key technology in the Internet of Things (IoT) [6–9]. With the extensive use of RFID technology, the privacy and security of RFID systems have also drawn great attention [10–21]. The type of privacy considered in this paper is mainly the privacy of the location of electronic tags and the privacy of the electronic tags' carrier. The type of security we focus on in this paper mainly includes the denial of unauthorized access to electronic tags' storage and the forgery of electronic tags. Protocol design is a common and effective method to solve the problem of privacy and security in RFID systems.

To popularize RFID technology, the price of electronic tags cannot be too high. The storage capacity and computing power of electronic tags are also limited, and traditional encryption algorithms cannot be applied directly in RFID systems. Thus, new protocol mechanisms must be designed. The formal verification of protocols and the discovery of protocol vulnerabilities have also become hot research topics. Very recently, Tewari and Gupta proposed an ultralightweight mutual authentication protocol in IoT environments for RFID tags that aims to provide secure communication with the least cost in both storage and computation, but this protocol is still vulnerable to key disclosure attacks [22]. Formal verification and analysis of the protocol can help us to identify flaws in the protocol and improve the security level of RFID systems. At present, there are some formal

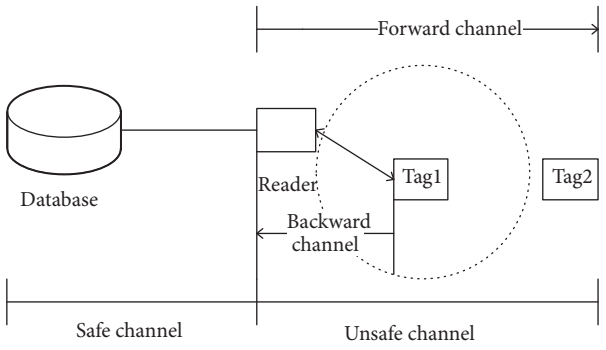

Figure 1: The RFID system.

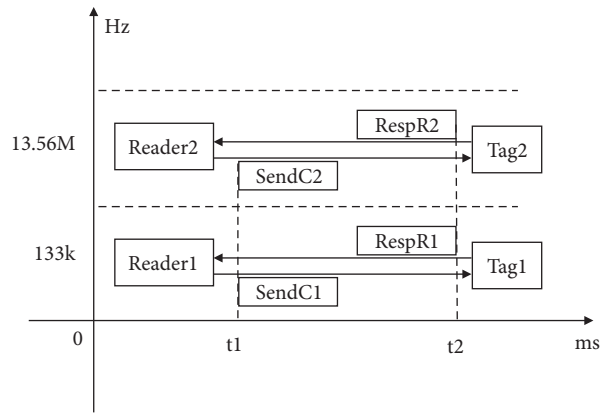

Figure 2: Anticollision protocol between electronic tags.

verification and analysis methods, such as the strong privacy model proposed by Jules [1, 23] and the random graph model proposed by Chang S [24]. The models mentioned above can be classified into two methods: enumeration methods (such as a random graph model) and induction methods (such as a strong privacy model). However, each formal model has shortcomings; for example, even a random graph model has advantages, such as being intuitive to use and quantitatively analysable. When this model uses a directed graph to search for a damage path, the random graph model is essentially an enumeration method.

When communication and the range of random numbers increase at the same time, the number of states in a random graph model will increase exponentially, and the state space will increase tremendously. A strong privacy model reduces the capability of adversaries and limits unauthorized communication by introducing parameters and employing a verification algorithm that is designed according to IND-CPA [25].

However, a strong privacy model still has the problem of a limited verification scale, and such a model will fail when the number of steps in communication is infinite. At the same time, the verification algorithm designed by Jules et al. is mainly used to verify whether the protocol is able to resist a denial of service attack, but it does not consider replay attacks.

We use a mathematical induction method to design a validation model to verify whether the protocol is able to resist a replay attack, and we formally verify the OSK protocol and VOSK protocol. We attempt to expand the verification scale, and the protocol can resist a denial of service attack when the number of steps in communication is infinite.

The paper is organized as follows. In Section 2, we present the RFID system model, and Section 3 describes the inductive model. In Section 4, we present the combination of inductive model and privacy algorithm on RFID systems. In Section 5, we use our method to verify the privacy and security of the OSK protocol and VOSK protocol. Conclusions are presented in the last section.

2. Preliminaries

RFID systems consist of three parts: electronic tags, a reader, and a background database, as shown in Figure 1.

Electronic tags can be divided into two major categories: active tags and passive tags. Active tags contain a battery and have a communication distance that is longer than that of passive tags. The lifetime of active tags is usually approximately 10 years. Passive tags have no battery inside, and the communication distance is shorter. Passive tags, the lifetime of which is not limited, can be activated by capturing the signal of the reader. The number of readers can be flexibly set for different applications, and readers can communicate with electronic tags within the scanning frequency range. The anticollision protocol between electronic tags is achieved in the physical layer, so the anticollision protocol between electronic tags in this paper is considered to be realized by default because it is achieved in the upper level. The anticollision protocol between electronic tags is shown in Figure 2.

3. Inductive Model

Electronic tags usually have a certain storage capacity and computing power. Electronic tags and readers need a two-way authentication protocol (some protocols only achieve one-way authentication). Active electronic tags can be read and written with an authorized reader, while passive tags are read-only. A reader with permission can read the memory of an electronic tag. The RFID systems considered in this section include a reader R and n electronic tags. Each participating entity in the RFID system can be considered an independent probability Turing machine; that is, the output value of the electronic tags and readers is random and is determined by the input and a local random number. Electronic tags and readers can be considered separate functional functions, and functional functions have well-defined interfaces that can receive messages from both parties or can respond to messages received.

3.1. Process of Modelling an Electronic Tag. Electronic tags typically store information such as the $tagId$, initialization key $s_{t,1}$, session identifier sid, and counter j. With the $SetKey$ message, the electronic tag can be assigned a new key value. After the electronic tag receives the $SetKey$, it will hand over

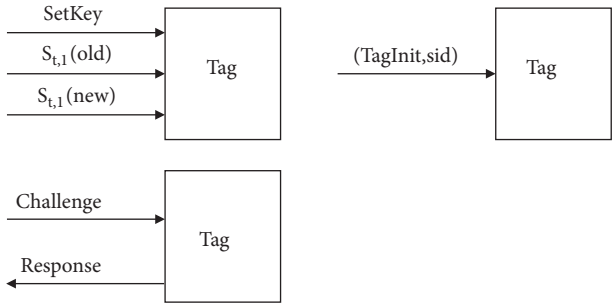

Figure 3: The electronic tag model.

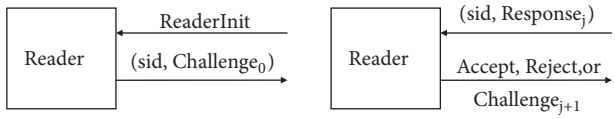

Figure 4: The formalized model of the reader.

its current key value and then randomly receive a new key value. If the electronic tag receives the message (*TagInit*, *sid*), the electronic tag will be assigned a new session identifier. At the same time, the *TagInit* message deletes all the information associated with the previous session identifier *sid*. Electronic tags can participate in only one protocol session at a time. The electronic tag responds to the challenge message generated by the reader, generating a response message. The function model of the electronic tag is shown in Figure 3.

3.2. Process of Modelling Readers. The inductive model proposed in this paper considers readers and the background database as a whole (a background database is usually a server, and the storage capacity and computing ability of this server can meet the needs of complex encryption algorithms. Thus, the communication channel between the reader and the background database can be regarded as a secure channel). RFID systems usually use symmetric encryption algorithms. The execution of the protocol is usually started by a reader, and the authentication result will be output by the reader. The reader will refresh the related data in the background database. The type of privacy indicated in the abstract mainly includes the position of the electronic tag carrier and personal information. No personal information is stored in the rom of electronic tags; rather, all personal information is stored in the background database. Only when the reader conducts the identity authentication of the electronic tag will the background database return the specific items of the reader. The model of the reader is shown in Figure 4.

Readers usually stay in a waiting state (waiting to scan electronic tags within the frequency range). When a reader receives the *ReaderInit* message, it will change to a working state and produce a challenge message $Challenge_0$ and session identifier *sid*. Then, the counter j will be reset, and $Challenge_0$ will be sent to the tag at the same time. The reader creates a list of sessions for each incoming message *ReaderInit*. The format is shown in the following:

$$(sid, \text{"}open\text{"}, Challenge_0, \ldots) \quad (1)$$

All the challenge messages $Challenge_i$ $(i = 1, 2, 3, \ldots)$ and the response messages $Response_i$ generated by the *sid* subsession are added to Equation (1) in order. Some of the secondary data generated in this session will also be added to the corresponding session list. When the session ends, "*Closed*" is added at the end of Equation (1). After the session list is added to "*Closed*," the session list goes into read-only mode. The reader will compute the next challenge message according to the protocol after receiving the $(Reader_i, Tag_t, Response_j, j)$ message. The reader may end authentication and output the result, either accepting or rejecting and updating the counter $j' = j$ and sending the updated value of the counter j together with the challenge message $challenge_j$ to Tag_t.

3.3. Adversary Modelling Process. Attackers not only can intercept messages on both the forward channel and the backward channel but also can send messages to the electronic tag on the forward channel and to the reader on the backward channel [1, 23, 24, 26, 27]. The model of the adversary is shown in Figure 5. The adversary can access the interface provided by the reader and tag; that is, the adversary can send some of their own messages:

$$SetKey, TagInit, ReaderInit, Challenge, Response \quad (2)$$

The calculation ability of the adversary is assumed to be bounded by polynomials. To formalize the calculation ability of the adversary, some parameters need to be introduced. Let r represent the number of *ReaderInit* messages sent by the adversary. s stands for the maximum number of communication steps, and t represents the number of messages sent by the adversary to the *TagInit* message. Because the number of challenge and response messages sent by an adversary is determined by r, s, and t, there is no explicit reference to the number of references to the challenge or the number of response messages. The *SetKey* message can obtain the key value inside the electronic tag and then randomly assign a new key value to the electronic tag, which essentially destroys the electronic tag. The adversary can use the *SetKey* message any time before it reaches s. In an RFID system environment with n electronic tags, it is possible to use *SetKey* messages for $n - 2$ electronic tags. We consider a nontrivial attack, that is, an attack in which at least two electronic tags are intact.

4. The Combination of Inductive Model and Privacy Algorithm on RFID Systems

The verification algorithm presented in this paper is based on the strong privacy model proposed in a previous article [23] and improves on the verification algorithm for the strong privacy model. The primary differences are as follows.

(1) The environment considered in this paper is one reader and n electronic tags; (2) the inductive model proposed in this paper can deal with a case in which the number

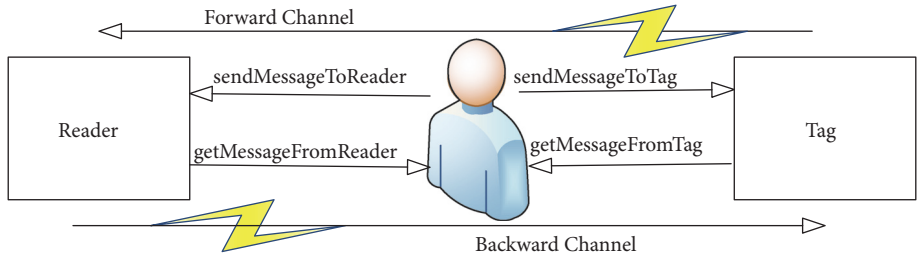

Figure 5: Formalization of the adversary.

of communication steps is infinitely large, but the strong privacy model proposed in [23] can deal with only a situation in which the communication has an upper limit m (m is a given constant), and the strong privacy model will fail when m is infinite; and (3) the inductive model can use mathematical induction to expand the verification scale.

4.1. Distinction between Two Electronic Tags. Because the unique identifier id of the message and the response from the electronic tag do not matter, the adversary can intercept only the response message set to derive the characteristics of the electronic tag. For inverse reasoning by an adversary to be successful, the response message $Response(Tag_t, k)$ of Tag_t and Tag_k must be paired successfully by the adversary, as shown in the following:

$$\begin{aligned} &(Tag_t, Response(Tag_t, 1)), \\ &(Tag_t, Response(Tag_t, 2)), \\ &\vdots \\ &(Tag_t, Response(Tag_t, p)) \end{aligned} \tag{3}$$

Among the above,

$$Response(Tag_t, k) = (Reader_i, Tag_t, Response_k, k) \quad (k = 1, 2, \ldots, \infty) \tag{4}$$

An adversary may extract characteristic information after collecting p pairs of two tuples according to Equation (3). If an authentication protocol can prevent an adversary from successfully matching or can reduce the probability of a successful match (the pairing success probability can be neglected for adversaries whose computing power is polynomially bounded), we can view the authentication protocol as having privacy. The model only needs to verify whether the protocol achieves indistinguishability between two electronic labels because the probability of an adversary matching p pairs of two tuples will not be more than $1/2^p$.

In only two electronic tags, if the probability of the opponent successfully matching the two tuples is not more than 1/2, 1/2 can be neglected for adversaries whose computing power is polynomially bounded. However, the probability of p pairs of two tuples existing will be less than 1/2 in the situation in which there are three electronic tags. In contrast, if a protocol has privacy in the environment with three electronic tags, the protocol cannot be guaranteed to maintain the privacy of the two electronic tags. Additionally, if a protocol maintains the privacy of the two electronic tags, it has the same privacy in the n environment. Thus, we simply need to verify that the protocol has achieved indistinguishability between two electronic tags.

The RFID system security issue is that legitimate electronic tags should be accepted by the reader, and an electronic tag forged by an adversary should be rejected.

4.2. Measuring the Advantage of an Adversary in a Challenge Game. The privacy experiments are based on the classic IND-CPA (choose plaintext attack security) and IND-CCA (choose ciphertext attack security) in cryptography. In this experiment, under the constraint of parameters r, s, and t (the computational power of polynomials) if the adversary has no significant advantage, the RFID protocol can be considered to have privacy.

The adversary's goal is to identify two different electronic labels. The adversary can access an interface that is limited to an electronic tag, and reader functions provide the access interface, but the adversary's computing power is limited.

A public key cryptosystem requires three algorithms, namely, an encryption algorithm, a decryption algorithm, and a key generation algorithm. The key generation algorithm is responsible for generating the key (public key, private key), using the public key encryption algorithm to encrypt plaintext and generate ciphertext, and using the private key to decrypt the ciphertext to recover the plaintext.

The encryption system considered in this paper is a probabilistic encryption system because the encryption algorithm is not able to completely eliminate the probability that the adversary will obtain some useful information or all useful information. The IND-CPA security concept was proposed by Goldwasser and Micali [25] and is the standard security level in public key cryptography. IND-CPA security, also called semantic security, as noted by Goldwasser and Micali, does not have to restore the whole plaintext attack on ciphertext, even with the recovery of one bit of information. This approach should also be considered an instance of cipher attack success.

For IND-CPA security, Avoine, Jules et al. designed (r, s, t)-privacy experiments and intersecting (r, s, t)-privacy experiments in their proposed enemy model and strong privacy model [1, 23, 26]. We will design (r, s, t)-security experiments on the basis of literature [16–18] and use (r, s, t)-privacy experiments and (r, s, t)-security experiments to

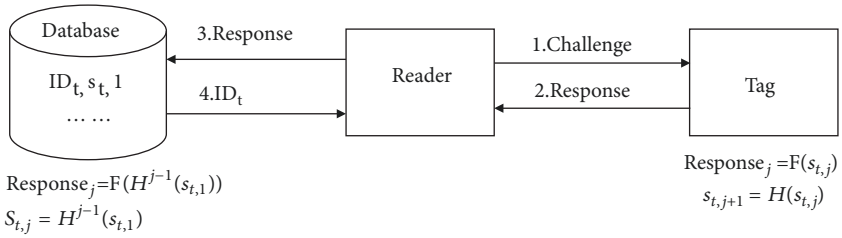

Figure 6: OSK protocol.

analyse the privacy and security of the OSK protocol and VOSK protocol.

Definition 1 (RFID system (r, s, t)-privacy). In the RFID system $S = (generate(k), R, Tag_1, Tag_2, \ldots, Tag_n)$, if the formula privacy is established, the protocol mechanism used in the communication between the reader and the electronic tag satisfies the (r, s, t)-privacy:

$$\forall r, s, t \in Z, \ \max \quad (r, s, t) < 2^k, \quad \text{s.t.} \quad p\left(x = x'\right) \leq \frac{1}{2} + \frac{1}{ploy\,(k)} \tag{5}$$

The challenge game based on IND-CPA standards [16–18] is as follows.

(1) Initialization phase:
The system generates n initialization key values $(s_{t,1}(tag_1), s_{t,2}(tag_2), \ldots, s_{t,n}(tag_n))$ and distributes the n key values to the n tags. Two electronic tags, tag_i, tag_j, are chosen randomly from the n electronic tag, presented to the opponent and shielded from the rest of the $n - 2$ tags.

(2) Learning phase:
The adversary begins communication between *reader* and tag_i, tag_j and intercepts a message on the forward channel and backward channel. The adversary will also use the forward channel to send a message to the electronic tag and use the backward channel to send a message to the reader. According to the protocol mechanism, the response message (*challenge*, *response*) is calculated, but the number of steps in the communication should not exceed s (s stands for the maximum number of steps).

(3) Challenge phase:
Assume that $Tag_0{}^* = Tag_i$, $Tag_1{}^* = Tag_j$ and that the prediction machine chooses a tag from tag_i, tag_j and signs it as $Tag_x{}^*$. The adversary begins communication with $Tag_x{}^*$. The adversary can intercept messages from the forward channel and backward channel. The adversary outputs his guess tag_x, and if $tag_x = Tag_x{}^*$ is successful, the RFID protocol is not private. Probability p ($tag_x = Tag_x{}^*$) is usually used to describe the enemy's ability.

5. Performance Evaluation

In this section, we will use our method to verify the OSK protocol and VOSK protocol. First of all we verify the privacy of the two protocols and then verify the security of VOSK protocol.

5.1. Verifying the Privacy of OSK Protocol and VOSK Protocol. In Figure 6, functions F and H are irreversible hash functions. Function F is used to compute the response message, and function H is used to refresh keys. The initial challenge message contains nothing, and it is used to start the session between electronic tags and readers. The value of this message will not change in the latter communication. The reader will search all $id_t, s_{t,1}$ in the background database after getting the response message:

$$Response = F\left(H^{j-1}\left(s_{t,1}\right)\right) \tag{6}$$

If Equation (6) is correct, the tag is legal, and the database will return unique identifier ID_t to the reader. The OSK protocol usually has a limited number of communication steps, for example, s.

According to this feature, the probability of success is 1. That is, $p(x = x') = 1$. Essentially, the adversary is still performing a denial of service attack. In fact, in the experiment, the adversary can make Tag_i reach the maximum number s of communication, so if the adversary chooses tag_i, then the reader will reject the electronic tag. tag_j is a legitimate electronic tag, and, in accordance with the normal protocol mechanism, the reader will receive the electronic tag.

The VOSK protocol has the same problem, although a VOSK protocol that introduces random numbers (nonce) can resist the replay attack. The VOSK protocol still has a communication count limit, so the adversary can still perform denial of service attacks.

In [23], Jules proved that the OSK protocol cannot defend against denial of service attacks using a strong privacy model when communication has an upper limit s. We will prove that the OSK protocol and the VOSK protocol can defend against denial of service attacks when the number of communication steps is not limited. We need to prove only that adversaries do not have significant advantages in the challenge game according to Definition 1. We make the initial phase and learning phase start according to Equation (6). Adversaries can send *sendNumber* challenge messages in the learning phase.

We assume that the adversary sends $sendNumber_i$ challenge messages to Tag_i and $sendNumber_j$ challenge messages to Tag_j. Additionally, $sendNumber_i + sendNumber_j < sendNumber$. And the adversary received $sendNumber_i$ response messages from Tag_i which can be written as message set $getMessageSet_i$, $|getMessageSet_i| \leq sendNumber_i$ and the same as $sendNumber_j$, response messages from Tag_j, $|getMessageSet_j| \leq sendNumber_j$. Then $getMessageSet = getMessage_i \cup getMessageSet_j \leq sendNumber$.

Effective response messages in the challenge phase consist of all legal messages from Tag_i, Tag_j and are written as $validResponseSet$. The set of legal response messages approaches infinity $|validResponseSet| \longrightarrow \infty$ because electronic tags can communicate an infinite number of times. The probability of an adversary guessing $p(x = x')$ correctly is presented in the following:

$$p\left(x = x'\right) \leq \frac{1}{2} + \frac{\left|getMessageSet\right|}{\left|validResponseSet\right|} \longrightarrow \frac{1}{2} \tag{7}$$

When $sendNumber$ increases by one, we obtain the following:

$$p\left(x = x'\right) \leq \frac{1}{2} + \frac{\left|getMessageSet + 1\right|}{\left|validResponseSet\right|} \longrightarrow \frac{1}{2} \tag{8}$$

Equation (7) means that when the number of communication steps is unlimited, the OSK protocol and VOSK protocol achieve indistinguishability between two electronic tags, but the OSK protocol cannot defend against replay attacks. We will also prove that the OSK protocol cannot defend against replay attacks. For example, the adversary can reply to any intercepted response message correctly, which means that electronic tags can be forged by adversaries.

5.2. Formal Verification Algorithm for Security. Compared with the privacy verification algorithm, the verification algorithm designed by this paper based on the (r, s, t)-privacy experiments is mainly used to validate whether the protocol can defend against replay attacks and to measure the advantage of an adversary in challenge games.

If the adversary cannot distinguish between the forged tag and the legal tag, then the adversary cannot forge a message received by the reader, which means that the adversary cannot successfully mount a replay attack.

(1) Initialization state: (1) Submit the *ReaderInit* message to the reader and allow it to change to a working state from a waiting state. (2) The system generates n initial key values and sends these key values to the electronic tags. (3) Choose two electronic tags tag_i, tag_j randomly from n electronic tags and present them to the adversary. Then, shield the other $n - 2$ electronic tags.

(2) Learning state: (4) The adversary begins communication between *reader* and tag_i, and tag_j intercepts messages on the forward channel and backward channel. The adversary may send messages to the electronic tags using the forward channel and backward channel. (5) The adversary sends m message *TagInit* to the electronic tag tag_i.

(3) Challenge state: (6) Assume that $Tag_0{}^* = Tag_i$ and $Tag_1{}^* = Tag_j$. The prediction machine chooses a tag from Tag_i, Tag_j and writes it as $Tag_x{}^*$. (7) The following three methods can be used in any order in the process of communication between the adversary and the electronic tag $Tag_x{}^*$.

(a) Send message *ReaderInit*, but do not send more than r times.

(b) Send message *TagInit*, but do not send more than t times.

(c) According to the protocol mechanism, the response message challenge and response are calculated, but the number of steps of the communication should not exceed s.

(8) The system generates an electronic tag *Bob*, which is a copy of $Tag_x{}^*$ and has the same computing power as $Tag_x{}^*$. *Bob* does not know the key value or the other data, but it can communicate with readers normally according to the protocol. (9) The adversary begins communication between the *reader* and *Bob* and can intercept the message on the forward channel and backward channel.

If the probability of an adversary distinguishing between *Bob* and Tag_x* is no more than 1/2, the adversary does not have advantages in the challenge game.

Definition 2. If the probability of an adversary distinguishing between *Bob* and $Tag_x{}^*$ in the challenge game is less than 1/2, the given protocol can defend against replay attacks and process the (r, s, t)-security.

5.3. Verifying Security of the VOSK Protocol. In Figure 7, there are two main differences between the OSK protocol and the VOSK protocol. First, the VOSK protocol sends a random number (*nonce*) to the electronic tag when it sends the challenge message. The electronic tags use a hash function to compute the key value, and the random number is received from the reader at the same time. Second, the background database refreshes the key with the two tuples, which meets the conditions when the background database is ergodicing the two tuples. According to the security experiment, we can get the following theorem.

Theorem 3. *The VOSK protocol can defend against replay attacks.*

Proof. (1) The probability of successfully replaying a legal response message is negligible. The legal response message of the replay attack is analysed first. A legitimate electronic tag responds to the reader for the first j times, and the inner state of the electronic tag is now $s_{t,j} = H^{j-1}(s_{t,1})$. The electronic tag produces response message $Response_j = F(s_{t,j} \mid nonce_j)$ and refreshes the key when it receives a challenge message $challenge_j, nonce_j$ from the reader. Then, the electronic tag sends the response message to the reader. If it is the first time that the electronic tag has sent this response message, the

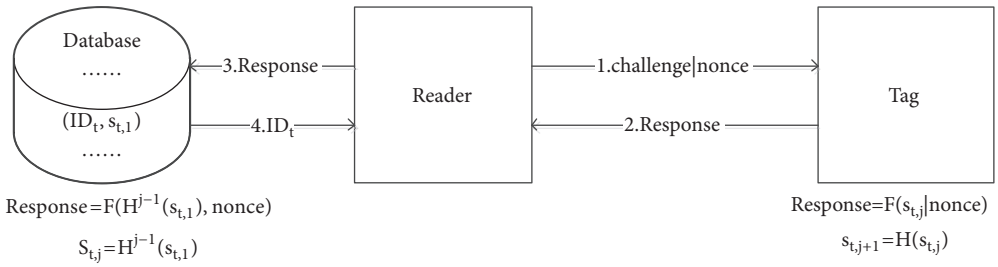

Figure 7: VOSK protocol.

reader accepts this electronic tag, the background database changes its two tuples from $(ID_t, s_{t,1})$ to $(ID_t, s_{t,j})$, and the reader changes the random number from $nonce_j$ to $nonce'_j$. The adversary can replay the message after copying it: first, the adversary sends $Response_j$ to the reader at the $k + j$ response in this session. The reader looks up the two tuples, which meets the following condition:

$$Response_j = F\left(H^{j+k-1}\left(s_{t,1}\right), nonce_j\right) \tag{9}$$

Because $H^{j+k-1} = s_{t,j+k} \neq s_{t,j}$ is permanent, there exist no two tuples that meet the condition of the equation, which means that the adversary cannot successfully replay the message in this session. Second, the adversary sends R$esponse_j$ to the reader for the first j responses in the next session, and the reader then looks up the two tuples, which meets the condition of

$$Response_j = F\left(H^{j-1}\left(s_{t,1}\right), nonce'_j\right) \tag{10}$$

If the equation is correct, $nonce_j = nonce'_j$ is correct as well. Assuming that the bit width of the random number is m, the probability of replaying a message successfully is $1/2^m$.

(2) The probability of the adversary successfully replaying the message with polynomial computing power is negligible. We consider the probability of the adversary successfully replaying the message with polynomial computing power in the following, namely, the probability of the adversary distinguishing between *Bob* and Tag_x^*. We take the initial state and learning state in order according to the challenge game. (1) The response message from tag_i, tag_j in the learning state is a legal message that is signed with message set $getMessageSet_1$. Because communication with the tag cannot exceed s, $getMessageSet_1 \leq s$. We know that the probability of replaying the message in $getMessageSet_1$ is zero and that the probability in the next turn is less than $getMessageSet_1/2^m \leq s/2^m$. (2) The adversary may generate its response message without using the response message set received in the learning state, and the format of the response message is $Response_j = F(s_{t,j} \mid nonce_j)$. The adversary can successfully replay the message only when both the values of the key and the random number are correct. (2) The adversary may generate its response message without using the response message set received in the learning state, and the format of the response message is $Response_j = F(s_{t,j} \mid nonce_j)$. The adversary can successfully replay the message only when the values of both the key and the random number are correct. Because the adversary cannot try more than r times *ReaderInit* and t times *TagInit*, the probability of the adversary successfully replaying the message in this condition is $(r+t)/2^m$, which means that the probability of the adversary distinguishing between *Bob* and Tag_x* is less than $(r + t + s)/2^m$.

Synthesizing (1) and (2) above, it can be seen that the probability of success of the opponent's replays is less than or equal to $(r+t+s)/2^m$, which is negligible for the opponent with a bounded polynomial. Synthesizing (1) and (2) above, it can be seen that the probability of success of the opponent's replays is less than or equal to $(r+t+s)/2^m$, which is negligible for an opponent with a bounded polynomial. The VOSK protocol can resist replay attacks with (r, s, t)-security. □

Through (r, s, t)-security experiments, this section verifies that the OSK protocol and VOSK protocol can resist a denial of service attack under the condition that there is no communication limit. The security experiment is designed on the basis of (r, s, t)-security experiments and proves that the VOSK protocol can resist message replay attacks.

6. Conclusion

We design a verification algorithm based on the privacy verification algorithm and combine it with mathematical induction. This method differs from that of Jules [23] in the fact that, when faced with a case in which the upper limit of steps is infinite, the model introduced by Jules is not applicable, while the induction-based model introduced in this work is applicable. The method of combining mathematical induction and model detection is valid only for property verification with inductive properties.

Future work will focus on the privacy and security of the general nature validation methods and the correlation between electronic label outputs.

Data Availability

In this paper, the privacy and security of RFID OSK protocol are mathematically verified by using formal verification method, and there is no specific experimental data.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grants nos. 61572109, 11461006, and 61772006; the Special Fund for Scientific and Technological Bases and Talents of Guangxi under Grant no. 2016AD05050; and the Special Fund for Bagui Scholars of Guangxi.

References

[1] A. Juels, "RFID security and privacy: a research survey," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 381–394, 2006.

[2] R. Want, "An introduction to RFID technology," *IEEE Pervasive Computing*, vol. 3, no. 2, pp. 183–186, 2010.

[3] K. Finkenzeller, *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*, John Wiely & Sons Ltd, Second edition, 2003.

[4] J. Landt, "The history of RFID," *IEEE Potentials*, vol. 24, no. 4, pp. 8–11, 2005.

[5] H. Ma, Y. Wang, and K. Wang, "Automatic detection of false positive RFID readings using machine learning algorithms," *Expert Systems with Applications*, vol. 91, pp. 442–451, 2018.

[6] J. P. van Straalen, A. Leyte, J. A. Weber et al., "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[7] S. Amendola, R. Lodato, S. Manzari, C. Occhiuzzi, and G. Marrocco, "RFID technology for IoT-based personal healthcare in smart spaces," *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 144–152, 2014.

[8] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): a vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[9] D. Miorandi, S. Sicari, F. de Pellegrini, and I. Chlamtac, "Internet of things: vision, applications and research challenges," *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497–1516, 2012.

[10] C. Kessler G, "An Overview of Cryptography," *Advanced Security & Privacy for Rfid Technologies*, 2016.

[11] M. R. Rieback, B. Crispo, and A. S. Tanenbaum, "The evolution of RFID security," *IEEE Pervasive Computing*, vol. 5, no. 1, pp. 62–69, 2006.

[12] K. Osaka, T. Takagi, K. Yamazaki, and O. Takahashi, "An efficient and secure RFID security method with ownership transfer," in *Computational Intelligence and Security*, vol. 4456 of *Lecture Notes in Computer Science*, pp. 778–787, Springer, Berlin, Heidelberg, 2007.

[13] S. Karthikeyan and M. Nesterenko, "RFID security without extensive cryptography," in *Proceedings of the ACM Workshop on Security of Ad Hoc and Sensor Networks, Sasn 2005*, pp. 63–67, Alexandria, Va, USA, November 2005.

[14] M. R. Rieback, G. N. Gaydadjiev, B. Crispo et al., "A platform for RFID security and privacy administration," *Lisa Usenix*, vol. 4, no. 4, pp. 70–72, 2006.

[15] B. Song and C. J. Mitchell, "Scalable RFID security protocols supporting tag ownership transfer," *Computer Communications*, vol. 34, no. 4, pp. 556–566, 2011.

[16] S. A. Weis, "RFID security and privacy," *Book of Extended Abstracts 11*, vol. 2, no. 2, pp. 48–50, 2005.

[17] S. E. Sarma, S. A. Weis, and D. W. Engels, "RFID systems and security and privacy implications," in *Cryptographic Hardware and Embedded Systems—CHES 2002*, pp. 454–469, Springer, Berlin, Germany, 2002.

[18] S. A. Weis, S. E. Sarma, R. L. Rivest, and D. W. Engels, "Security and privacy aspects of low-cost radio frequency identification systems," in *Security in Pervasive Computing*, Lecture Notes in Computer Science, pp. 201–212, Springer, Berlin, Heidelberg, Germany, 2004.

[19] R.-I. Paise and S. Vaudenay, "Mutual authentication in RFID: security and privacy," in *Proceedings of the 2008 ACM Symposium on Information, Computer and Communications Security, ASIACCS '08*, pp. 292–299, March 2008.

[20] S. Garfinkel and B. Rosenberg, *Applications, Security, And Privacy*, Pearson Education India, 2006.

[21] D. Molnar and D. Wagner, "Privacy and security in library RFID issues, practices, and architectures," in *Proceedings of the 11th ACM Conference on Computer and Communications Security, CCS 2004*, pp. 210–219, USA, 2004.

[22] K. H. Wang, C. M. Chen, W. Fang, and T. Y. Wu, "On the security of a new ultra-lightweight authentication protocol in IoT environment for RFID tags," *The Journal of Supercomputing*, vol. 74, no. 1, pp. 1–6, 2017.

[23] A. Juels and S. A. Weis, "Defining strong privacy for RFID," *ACM Transactions on Information and System Security*, vol. 13, no. 1, 2009.

[24] S. Chang, H. Song, L. Lu, Q. Yao, and Y. Qi, "Random graph based benchmarking methodology for RFID security protocols," in *Proceedings of the 2013 IEEE 10th International Conference on e-Business Engineering, ICEBE 2013*, pp. 184–191, UK, September 2013.

[25] C. Fontaine and F. Galand, "A survey of homomorphic encryption for nonspecialists," *EURASIP Journal on Information Security*, vol. 2007, 2007.

[26] G. Avoine, "Adversarial model for radio frequency identification," *Cryptology ePrint Archive*, vol. 49, 2005.

[27] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, "Strong authentication for RFID systems using the AES algorithm," in *Cryptographic Hardware and Embedded Systems - CHES 2004*, vol. 3156 of *Lecture Notes in Computer Science*, pp. 357–370, Springer, Berlin, Heidelberg, 2004.