Sensors and Materials, Vol. 31, No. 3 (2019) 969–979
MYU Tokyo

S & M 1828

Improved Privacy-preserving Authorized Out Authentication Protocols

Jinbin Zheng[1*] and Fangguo Zhang[2,3]

[1]School of Mathematics and Information Engineering, Longyan University, Longyan, Fujian, China
[2]School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China
[3]Guangdong Provincial Key Laboratory of Information Security, Guangzhou 510006, China

(Received November 23, 2018; accepted March 4, 2019)

Keywords: radio frequency identification (RFID), authorization, privacy-preserving authorization, authentication protocol

With wide applications of the radio frequency identification (RFID) technology in areas such as the supply chain, warehouse management, and so on, the privacy and security of RFID gradually become one of the hot topics. RFID privacy and security authentication protocols have been proposed to suit different scenarios. In 2014, the Alert Response Address (ARA) protocol based on the application scenario of a privileged membership club was presented, the formal definition of its privacy and security models was described, and the tag and reader's privacy were preserved. In this study, we analyze the efficiency of the ARA scheme and find that it is not very good because the scheme overuses bilinear maps. Under the premise of without losing the ARA scheme's privacy and security, we improve its two subprotocols, IARA1 and IARA2, with higher performance in the ARA protocol. The results show that the improved protocol is more efficient than the ARA protocol through the performance comparison of the protocols.

1. Introduction

Radio frequency identification (RFID) technology is widely used in today's intelligent logistics and management supply chain, manufacturing and assembly, ticket management and warehouse management, and other systems, and is a very common and effective technology. In the RFID system, users' personal information, hobbies, and the like can be linked by the unique identification code of the tags themselves, and the privacy leakage of the RFID system may be caused. At the same time, tags should not disclose any sensitive information to unauthorized readers, for which system security issues must be taken into account. Therefore, when designing and using the RFID system protocol, how to ensure that only authorized readers can identify legitimate tags and how to prevent attackers from malicious attacks, such as tracking, intercepting, and leaking to protect the privacy of users' information and ensure the security of the system, should be considered.

[*]Corresponding author: e-mail: jbzheng518@163.com

https://doi.org/10.18494/SAM.2019.2221

To resolve the privacy and security issues of the RFID system, many related RFID authentication protocols, such as LAMP,(1) LAMP + 1,(2) SASI,(3) Gossamer,(4) AFMAP,(5) and RAPP(6) protocols, have been proposed. These protocols are authenticated by the introduction of the IDS (index-pseudonym), which is used to store a corresponding index in the tag information record, and each tag is associated with a key K. These solutions are based on the assumption of a secure communication of a back-end information channel that can only be used to perform simple binary bit operations, and most of the solutions are applied to the security certification of lightweight and ultra-lightweight RFID systems. In 2008, Tan *et al.*(7) proposed an RFID security authentication protocol without a back-end database. In this protocol, it is assumed that the information channel between the reader and the certification agency (CA) is secure, and the reader must download the tag-related information from the third-party CA to the device in the form of the access list L. Then, the protection of private information of the tag in the list L is implemented by a one-way hash function, and the authentication of the tag is realized. The protocol realizes functions such as antiprivacy disclosure, anticloning, antitracking, and antieavesdropping. In the same year, Ahamed *et al.*(8) proposed a lightweight RFID security search protocol without a back-end database for readers to search for a specified tag in a set of tags. On the other hand,(9) Lee *et al.*(9) pointed out that Tan *et al.*(7) only realized the one-way authentication of the tag by the reader, not the authentication of the reader by the tag. Once the mobile terminals of readers are stolen, the tag information will face the threat of being leaked. Therefore, they proposed a protocol to enhance authentication security and pointed out that the protocol could also resist attacks, such as privacy disclosure, cloning, tracking, and eavesdropping, and has better security than the previous protocol. To reduce the calculated amount of the authentication protocol indicated in Ref. 7 to improve its efficiency and meet the same security requirements, Lin *et al.* proposed a lightweight RFID authentication and search protocol without a back-end database.(10) However, with the widespread use of the Internet of Things and the rapid development of cloud computing, cloud servers tend to replace RFID-specific back-end servers with their advantages of large storage capacity, abundant resources, and strong computing power. For example, Chen *et al.*(11) used a cloud database to reduce the complexity of search and avoid data inconsistency, thus proposing an RFID privacy protection authentication protocol. Kardas *et al.*(12) proposed a cryptographic protocol based on a symmetric key using cloud computing. In order to realize the identification of tags by fixed or mobile readers anytime and anywhere, Xei *et al.*(13) proposed a cloud-server-based RFID authentication protocol and pointed out that the protocol could provide large-scale background resource services with better robustness. Kiraz *et al.*(14) explored a two-way authentication protocol for RFID systems based on trusted cloud providers.

The corresponding RFID authentication protocols are required to be implemented in different application scenarios because of the different needs of enterprises. In 2014, Li *et al.*(15) proposed the Alert Response Address (ARA) protocol based on the application scenarios of club members' consumption and gave formal definitions of system privacy and security model, which realized the privacy protection of tags and readers. In this study, on the basis of the system model of the ARA protocol, the ARA protocol, referred to as the IARA protocol, was improved without any loss of privacy and security. The results showed that the improved IARA protocol had a higher operating efficiency by analyzing and comparing the efficiencies of the ARA and IARA protocols.

In Sect. 2 of this paper, we provide a brief introduction of the prerequisites for the privacy and security of the RFID system. In Sect. 3, we mainly describe the specific certification process of the ARA protocol proposed by Li *et al.*(15) In Sect. 4, we detail the authentication process for the improved IARA protocol proposed in this paper. In Sect. 5, we show that the IARA protocol is more efficient than the ARA protocol through performance analysis and comparison. Finally, in Sect. 6, we provide the conclusion.

2. Prerequisites

2.1 Composition and security model of RFID system

Usually, an RFID system generally consists of the following three major components: RFID tag (tag or transponder), RFID reader (reader or transceiver), and back-end database (back-end server),(16) as shown in Fig. 1. It is generally assumed that a tag does not have the ability to prevent tampering and destruction, and attackers can damage the tag and obtain all the secret information stored in the tag; at the same time, the tag has limited storage and computing resources. Public key cryptosystems generally do not apply to the tag, which can perform operations such as hash function and pseudorandom number generation. However, a reader is a device with relatively strong computing and storage capabilities, and can perform more complete cryptographic operations. The back-end server, as a storage device, stores all relevant information(17) about the tag in the system. In addition, in the design of an RFID system, the reader and back-end database are generally regarded as a whole, and the internal communication between them is assumed to be secure; on the other hand, the public wireless communication channel between the tag and the reader is unsafe, which may suffer from various active or passive attacks.

The security of an RFID system should meet the following requirements:

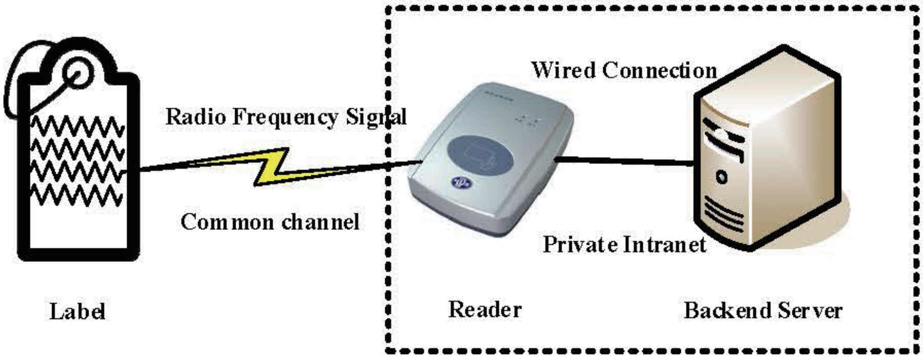

Fig. 1. (Color online) Components of RFID system.

1. Forward privacy: In a forward secure game, adversary *A* can obtain the current key of the reader and attempt to distinguish whether the currently authenticated tag has interacted with the server or other readers. If no adversary can win the game with the probability of $\frac{1}{2}+\varepsilon$ (ε is a negligible function), then the system is considered to meet forward privacy.
2. Backward privacy: In a backward security game, after the reader updates the key, adversary *A* attempts to determine whether any tag has interacted with the server or other readers after that. If no adversary can win the game with the probability of $\frac{1}{2}+\varepsilon$ (ε is a negligible function), then the system is considered to meet backward privacy.
3. Readers' privacy: In the privacy security game of the reader, adversary *A* can obtain the server's key and try to determine which registration tag is authenticated at the moment. If the probability that adversaries win the game is negligible, then the system is considered to meet readers' privacy.
4. Unforgeability security: In the unforgeability security game, adversary *A* can obtain the keys of the server and reader, and try to forge the output information of a certain tag for passing the authentication of the system during the authentication phase. If the probability that adversaries win the game is negligible, then the system is considered to be unforgeable or secure.

To describe the operations that adversary *A* and challenger *C* can perform in a secure game, Li *et al.* abstractly proposed a type of Oraclas program for calling[15] and gave a formalized definition of the RFID system security model. Detailed definitions and certifications are shown in Ref. 15.

2.2 Definition and properties of bilinear map

Suppose G and G_T are cyclic groups whose order is prime q, and g is the generator of group G; then, $\hat{e} : G \times G \rightarrow G_T$ is called a bilinear map, with only the following three properties:[18]

(1) Bilinearity: $\hat{e}\,(u^a, v^b) = \hat{e}\,(u, v)^{ab}$, in which $u, v \in G$, $a, b \in \mathbb{Z}_q$;

(2) Computability: For any $u, v \in G$, bilinear map $\hat{e}\,(u, v)$ is valid and computable;

(3) Nondegeneration: If g is a generator of G, then $\hat{e}\,(g, g)$ is a generator of G_T.

3. ARA Protocol Described by Li *et al.*

To protect the personal privacy of privileged members in the club, a new authorization protocol, referred to as the ARA protocol, was proposed.[15] The significant differences between this protocol and many of the previous authentication protocols are as follows: for consistency, reader *R* and background server *S* are relatively independent; reader *R* should be authorized by background server *S* before tags can be authenticated; in the process of reader *R* authenticating tag *T*, background server *S* does not know the information of tag *T*. Li *et al.*[15] divided the ARA protocol into three subprotocols P1, P2, and P3, according to the degree of protection of privacy information of tags; in this paper, we will not discuss P1 further because it only basically realizes the members' privacy protection, but it is mainly for the briefing and improvement of P2 and P3.

3.1 Algorithm description of ARA protocol

The authentication process of the ARA protocol mainly consists of four algorithms, as shown in Table 1.

3.2 P2 and P3 of the ARA protocol

The authentication of the P2 and P3 of the ARA protocol can be divided into two stages: The first stage is the tag registration or parameter initialization stage, that is, when the tag is registered, the server issues a series of public information such as the server public key, and the tag selects the required public information according to its own needs for initialization. When the membership card is activated, the public key of the tag is sent to the server, and the private key is reserved by the tag itself. Then, the server authorizes the reader, that is, the server assigns a deadline key to the corresponding reader, and once the deadline of the key has expired, the reader no longer has the authority of tag authentication; the second stage is that the reader authenticates the tag, in which the reader and server must work together to confirm whether the tag has been registered and legal.

In the P2 and P3 of the ARA protocol, a multiplicative cyclic group whose order is prime p is denoted by G and G_T; g, $h \in G$ are two different generators of G, and $x, \alpha \in Z_q^*$ are bilinear maps associated with the cyclic group. $\gamma, x, \alpha \in \mathbb{Z}^*_p$, T_R is a collection of tags. Assume that $H_2 : \{0,1\}^* \rightarrow \mathbb{Z}^*_p$ is a hash function that maps an arbitrary bit string to a multiplicative cyclic group in the number field $\mathbb{Z}_p$, and H_3: $\{0,1\}^* \rightarrow G$ is a hash function that maps an arbitrary bit string to the cyclic group G. The specific authentication processes for the P2 and P3 of the ARA protocol are shown in Figs. 2 and 3, respectively. The authentication steps are shown in Ref. 15.

4. Improved Protocol

Li *et al.* proposed the ARA protocol for the application scenario of a privileged member club,[15] which realized the privacy protection of authorized RFID. On the basis of the ARA

Table 1
Main description of ARA protocol.

Algorithm name	Input parameters	Output parameters
ServerKeyGen()	(k)	Generate public and private key pair of the server: (PK , SK).
TagkeyGen()	(t, k)	Generate public and private key pair of the tag: (pk, sk).
ReaderAuth()	({pki}, T_R, sk, R)	Generate public and private key pair of the reader: (rpk, rsk). Send rsk to the server and send rpk to the reader.
Auth()	(sk, PK, rsk, rpk)	The reader authenticates the tags; if passed, the corresponding tag identifier T is output; otherwise, the algorithm is terminated.

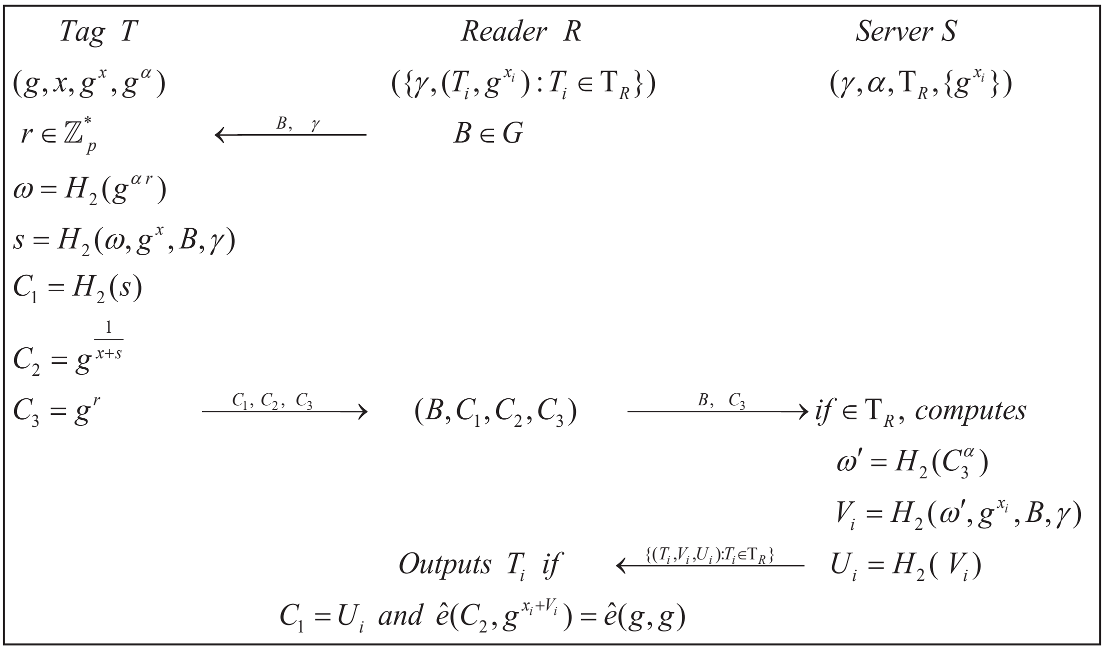
Tag T
Reader R
Server S
$(g, x, g^{x}, g^{\alpha})$
$(\{\gamma, (T_i, g^{x_i}) : T_i \in \mathrm{T}_R\})$
$(\gamma, \alpha, \mathrm{T}_R, \{g^{x_i}\})$
$r \in \mathbb{Z}_p^*$
B, γ
$B \in G$
$\omega = H_2(g^{\alpha r})$
$s = H_2(\omega, g^{x}, B, \gamma)$
$C_1 = H_2(s)$
$C_2 = g^{\frac{1}{x+s}}$
$C_3 = g^{r}$
C_1, C_2, C_3
(B, C_1, C_2, C_3)
B, C_3
if $\in \mathrm{T}_R$, *computes*
$\omega' = H_2(C_3^{\alpha})$
$V_i = H_2(\omega', g^{x_i}, B, \gamma)$
Outputs T_i *if*
$\{(T_i, V_i, U_i) : T_i \in \mathrm{T}_R\}$
$U_i = H_2(V_i)$
$C_1 = U_i$ *and* $\hat{e}(C_2, g^{x_i + V_i}) = \hat{e}(g, g)$

Fig. 2. Specific authentication process of ARA protocol 2.

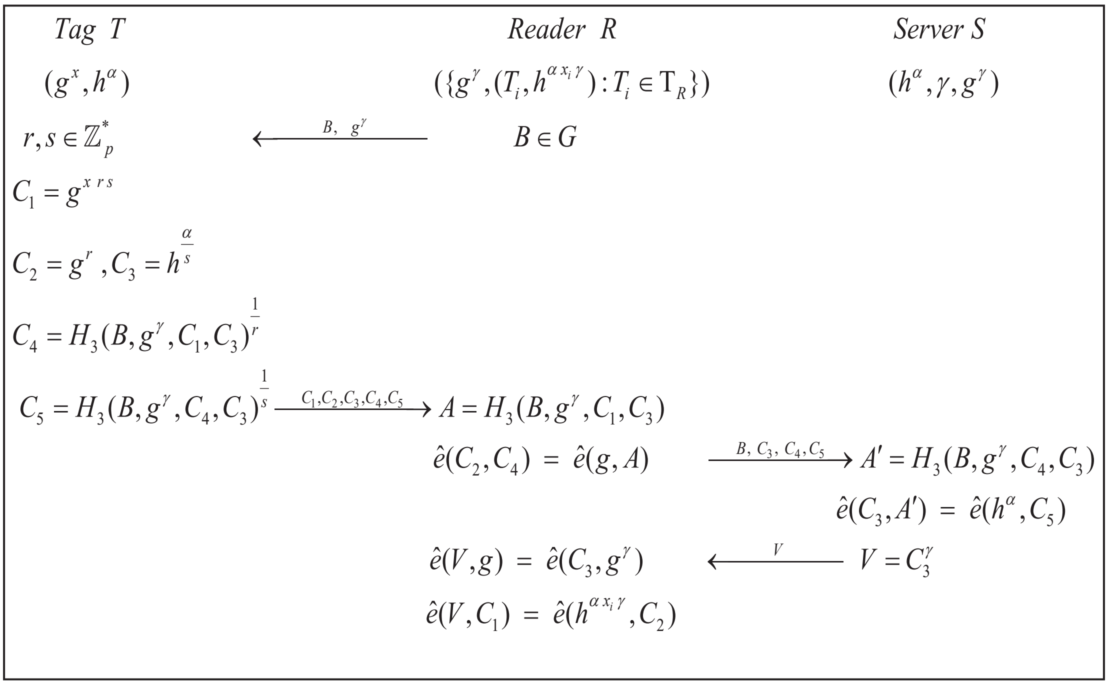
Tag T
Reader R
Server S
(g^{x}, h^{α})
$(\{g^{\gamma}, (T_i, h^{\alpha x_i \gamma}) : T_i \in \mathrm{T}_R\})$
$(h^{\alpha}, \gamma, g^{\gamma})$
$r, s \in \mathbb{Z}_p^*$
B, g^{γ}
$B \in G$
$C_1 = g^{x r s}$
$C_2 = g^{r}, C_3 = h^{\frac{\alpha}{s}}$
$C_4 = H_3(B, g^{\gamma}, C_1, C_3)^{\frac{1}{r}}$
$C_5 = H_3(B, g^{\gamma}, C_4, C_3)^{\frac{1}{s}}$
C_1, C_2, C_3, C_4, C_5
$A = H_3(B, g^{\gamma}, C_1, C_3)$
$\hat{e}(C_2, C_4) = \hat{e}(g, A)$
B, C_3, C_4, C_5
$A' = H_3(B, g^{\gamma}, C_4, C_3)$
$\hat{e}(C_3, A') = \hat{e}(h^{\alpha}, C_5)$
$\hat{e}(V, g) = \hat{e}(C_3, g^{\gamma})$
V
$V = C_3^{\gamma}$
$\hat{e}(V, C_1) = \hat{e}(h^{\alpha x_i \gamma}, C_2)$

Fig. 3. Specific authentication process of ARA protocol 3.

system model, under the premise of without losing its privacy and security, for the P2 and P3 of the ARA protocol, we proposed two more efficient optimization protocols in this paper, namely, the improved IARA protocols, referred to as IARA1 and IARA2.

4.1 Authentication process of IARA1

Let G, G_T be a cyclic group whose order is prime p, $x, \alpha \in Z_q^*$ a bilinear map associated with the cyclic group, and g the generator of G. H_1: $\{0,1\}^* \rightarrow \{0,1\}^n$ is a hash function (such as SHA-1) that maps an arbitrary bit string to a fixed bit string. The protocol process is described as follows.

4.1.1 Initialization phase

(1) *ServerKeyGen*(): Select $\alpha \in \mathbb{Z}_p^*$ randomly and calculate (PK, SK) = (g^α, α) as the public and private keys of server S;

(2) *TagKeyGen*(): Select $x \in \mathbb{Z}_p^*$ randomly and calculate (pk, sk) = (g^x, x) as the public and private keys of tag T; tag T is stored as (g, sk, pk, PK). The public key of server S for storing tag T is pk;

(3) *ReaderAuth*(): To authorize reader R to verify tag set T_R, server S randomly selects that and stores it as $\gamma \in \mathbb{Z}_p^*$. The reader stores rpk = (γ, T_R).

4.1.2 Stage of the reader authenticating tags

Algorithm *Auth*() for readers authenticating tags: During the execution of the algorithm, the specific process of interactive authentication among the tag, the reader, and the server is shown in Fig. 4.

4.2 Authentication process of IARA2

Following the symbolic description of the IARA protocol in the previous text, h is the generator of G and $h \neq g$. The authentication process of IARA2 is described as follows.

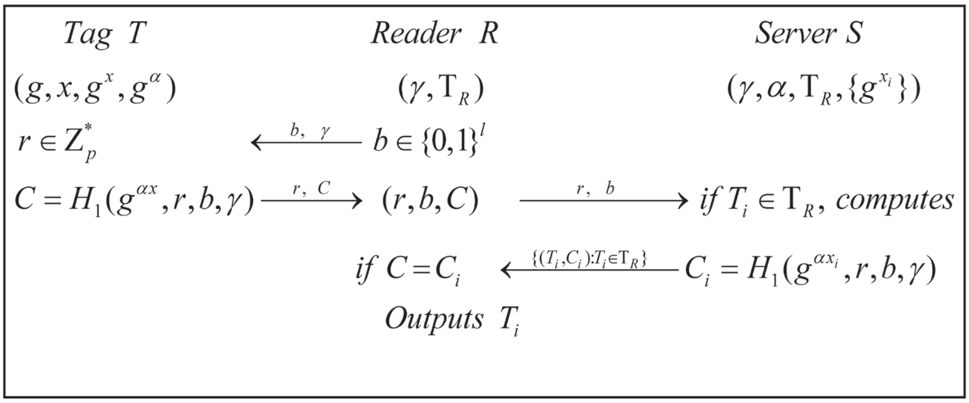
Tag T
Reader R
Server S
(g, x, g^x, g^α)
(γ, T_R)
$(\gamma, \alpha, \mathrm{T}_R, \{g^{x_i}\})$
$r \in Z_p^*$
$\xleftarrow{b,\ \gamma}$
$b \in \{0,1\}^l$
$C = H_1(g^{\alpha x}, r, b, \gamma) \xrightarrow{r,\ C}$
(r, b, C)
$\xrightarrow{r,\ b}$
$if\ T_i \in \mathrm{T}_R$, computes
$if\ C = C_i$
$\xleftarrow{\{(T_i, C_i): T_i \in \mathrm{T}_R\}}$
$C_i = H_1(g^{\alpha x_i}, r, b, \gamma)$
Outputs T_i

Fig. 4. The specific authentication process of IARA1 protocol.

4.2.1 Initiation phase

(1) *ServerKeyGen*(): Select $\alpha \in \mathbb{Z}_p^*$ randomly and generate the public and (*PK*, *SK*) = (h^α, α) private key pair corresponded by server *S*;

(2) *TagKeyGen*(): Randomly select the parameters $x \in \mathbb{Z}_p^*$ and generate the (*pk*, *sk*) = (h^x, g^x) public and private key pair corresponded by tag *T*. Tag *T* stores the information (*g*, *sk*, *pk*, *PK*) and server *S* stores the public key *pk* of tag *T*;

(3) *ReaderAuth*(): To authorize reader *R* to verify tag set T_R, server *S* randomly selects and stores $\gamma \in \mathbb{Z}_p^*$, and calculates g^γ. For each tag $T_i \in T_R$, the corresponding $(h^{x_i})^{\alpha\gamma}$ is calculated. Reader *R* stores $rpk = (g^\gamma, \{(T_i, h^{\alpha x_i \gamma}) : T_i \in T_R\})$.

4.2.2 Stage of tag authentication by the reader

Algorithm of the reader authenticating tag *Auth*(): The specific process of interactive authentication among the tag, the reader, and the server is shown in Fig. 5.

(1) Reader *R* randomly selects $b \in \{0,1\}^l$ and sends (b, g^γ) to tag *T*.

(2) Tag *T* randomly selects two values $r, s \in \mathbb{Z}_p^*$, figures out $C_1 = H_1(\hat{e}(g^\gamma, h)^s)$ and $C_2 = H_1(C_1, b, r, g^\gamma)$, and sends $(r, C_2, g^{\frac{s}{x}})$ as a response value to reader *R*;

(3) After reader *R* receives the tag feedback information $(r, C_2, g^{\frac{s}{x}})$, it sends $g^{\frac{s}{x}}$ to server *S*;

(4) Server *S* calculates $V = g^{\frac{s}{x\alpha}}$ and returns *V* to reader *R*;

(5) Reader *R* calculates and judges $H_3(H_3(\hat{e}(V, h^{\alpha x_i \gamma})), b, r, g^\gamma) = C_2$ one by one for the stored information $(T_i, h^{\alpha x_i \gamma})$. If there is a data pair $(T_i, h^{\alpha x_i \gamma})$ that satisfies the equation, indicating that there exists a tag that has been registered with server *S* and is legal, the corresponding legitimate tag identifier T_i is output; otherwise, the authentication process is terminated.

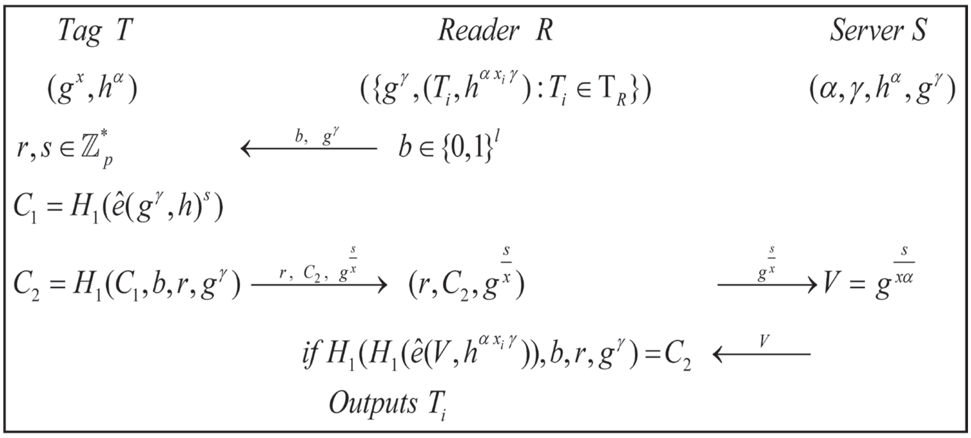
Tag T
Reader R
Server S
(g^x, h^α)
$(\{g^\gamma, (T_i, h^{\alpha x_i \gamma}) : T_i \in \mathrm{T}_R\})$
$(\alpha, \gamma, h^\alpha, g^\gamma)$
$r, s \in \mathbb{Z}_p^*$
$\xleftarrow{b,\ g^\gamma}$
$b \in \{0,1\}^l$
$C_1 = H_1(\hat{e}(g^\gamma, h)^s)$
$C_2 = H_1(C_1, b, r, g^\gamma) \xrightarrow{r,\ C_2,\ g^{\frac{s}{x}}} (r, C_2, g^{\frac{s}{x}})$
$\xrightarrow{g^{\frac{s}{x}}} V = g^{\frac{s}{x\alpha}}$
$if\ H_1(H_1(\hat{e}(V, h^{\alpha x_i \gamma})), b, r, g^\gamma) = C_2 \xleftarrow{V}$
Outputs T_i

Fig. 5. The specific authentication process of IARA2 protocol

5. Protocol Analysis and Comparison

Li *et al.* performed an efficiency analysis for the P2 and P3 of the ARA protocol.(15) To illustrate that the improved protocol in this paper is more efficient than the protocol proposed by Li *et al.* in this section, performance analysis, comparison, and security descriptions are performed for IARA1, IARA, P2, and P3.

5.1 Performance analysis and comparison

The operating efficiency of this protocol scheme is mainly reflected in the reader computing load and server computing load, and the communication traffic depends on the number of bits of data transmitted between the reader and the server. The following is a performance analysis for the calculated amount at the reader and server sides and the traffic between the reader and the server for IARA1 and IARA2:

Without loss of generality, we assume that the difference in operational efficiency between different hash functions is negligible, and in the efficiency analysis, the running time of any hash function is represented by H. G is used to indicate the time of the group operation. $\hat{e}$ is used to represent the operation time of the bilinear pair. The number of bits of any element X is represented by $|X|$.

1. Calculated amount at the reader side: In IARA1, the reader only needs to compare $C = C_i$ once for the C_i returned by the server, so the calculated amount of the reader is $O(1)$. Compared with the protocol P2, the reader needs to execute the group element and bilinear mapping operation, and the execution efficiency of the protocol reader is significantly improved.
2. Calculated amount at the server side: The information α and g^{x_i} have been prestored at the server side, and $g^{\alpha x_i}$ can be obtained by precalculation. When $C_i = H_1(g^{\alpha x_i}, r, b, \gamma)$ is calculated, only the H hash operation is completed once; thus, when the number of tags is n, the calculated amount of the server is nH. The execution efficiency is also significantly higher than $G + (2n + 1)H$ in the computation of the P2 of the ARA protocol.
3. Communication traffic: According to the assumption of the P2 of the ARA protocol, the reader has prestored the tag information $T_i \in T_R$, and the server does not consider the transmission of T_i when transmitting the data to the reader, so only the data set $\{C_i\}$ should be transmitted. When there are n tags, the traffic of the IARA1 is nba, which saves half of the data transmission amount compared with that of the P2, so IARA1 has a higher operating efficiency.

According to the efficiency analysis result of IARA1, the efficiency analysis result of IARA2 can also be obtained as well. The results of the efficiency analysis and comparison between the improved IARA protocol proposed in this paper and the ARA protocol proposed by Li *et al.* are shown in Table 2. The results of the analysis higher than that of the previous ARA protocol are shown in Table 2.

In fact, the performance analysis in Ref. 15 is incorrect, taking P2 as an example. The calculated amount of P2 should not be $G + 2nH$ at the server side but $G + (2n + 1)H$. The

Table 2
Performance comparison between ARA and IARA protocols.

Protocol	Calculation at reader side	Traffic	Calculation at server side
P2	$G + \hat{e}$		$G + (2n + 1)H$
P3	$(n + 5)\hat{e}$	$\|G\|$	$G + 2\hat{e}$
IARA1	$O(1)$		nH
IARA2	$n\hat{e}$	$\|G\|$	G

Table 3
Security analysis between ARA and IARA protocols.

Protocol	Forward privacy	Backward privacy	Reader privacy	Unforgeability of tags	Constant level communication cost	Constant level reader authentication cost	Tag performance
P2	√	√	√	√	×	√	H
P3	√	√	√	√	√	×	H, PK
IARA1	√	√	√	√	×	√	H
IARA2	√	√	√	√	√	×	H, PK

Note: √: may solve the security problem; ×: cannot solve the security problem; H: need to complete the hash operation; PK: need to complete the ECC operation.

calculated amount of P3 should not be $(n + 1)\hat{e}$ at the reader side but (n + 5)$\hat{e}$. The calculated amount of P3 should not be $G + \hat{e}$ at the server side but $G + 2\hat{e}$.

5.2 Security analysis

The ARA protocol constructs a privacy and security model based on the adversary challenge and Oracle machine definition, and assumes that the communication channel between the reader and the server is secure, and then analyzes that the protocol can prevent the leakage of the privacy information of tags and has forward privacy, backward privacy, and unforgeability of tags. Since the construction method for the privacy and security model of the IARA protocol is the same as that for the ARA protocol, specific proof is found in Ref. 15. Therefore, the IARA protocol has the same privacy and security as the ARA protocol, as shown in Table 3.

6. Conclusion

On the basis of the system model of the ARA protocol, we provide the definition of RFID system security in this paper. By describing the authentication processes of the P2 and P3 of the ARA protocol, we propose two subprotocols, IARA1 and IARA2, with higher performance, and calculate and analyze the calculated amount at the reader side, the calculated amount at the server side, and the traffic between the reader and server sides. Without losing its privacy and security, we compared and analyzed the performance and security before and after the ARA protocol was optimized. The results show that the improved IARA protocol is more efficient than the ARA protocol.

References

1 P. Peris-Lopez, J. C. Hernandez-Castro, and J. M. Estévez-Tapiador: Workshop on RFID Security (2006) 12.
2 T. Li: Vehicular Technology Conf. (IEEE, 2008) 1.
3 H. Chien: Dependable Secure Comput. **4** (2007) 337.
4 P. Peris-Lopez, J. C. Hernandez-Castro, and J. M. E. Tapiador: Information Security Applications: 9th Int. Workshop (2008) 23.
5 A. Sadighian and R. Jalili: Emerging Security Information, Systems and Technologies (IEEE, 2009) 31.
6 Y. Tian, G. Chen, and J. Li: IEEE Commun. Lett. **16** (2012) 702.
7 C. Tan, B. Sheng, and Q. Li: IEEE Trans. Wireless Commun. **7** (2008) 1400.
8 S. I. Ahamed, F. Rahman, and E. Hoque: Proc. 2008 2nd Int. Conf. Computer and Electrical Engineering (IEEE, 2008) 187.
9 C. F. Lee, H. Y. Chien, and C. S. Laih: Int. J. Commun. Syst. **25** (2012) 376.
10 I. C. Lin, S. C. Tsaur, and K. P. Chang: Proc. 2009 3rd Int. Conf. Computer and Electrical Engineering (IEEE, 2009) 295.
11 S. M. Chen, M. E. Wu, and H. M. Sun: J. Future Generation Comput. Syst. **30** (2014) 155.
12 S. Kardas, S. Celik, and M. A. Bingöl: IEEE 5th Int. Conf. Cloud Computing Technology and Science (CloudCom) (IEEE, 2013) 171.
13 W. Xie, L. Xie, and C. Zhang: IEEE Int. Conf. RFID (IEEE, 2013) 168.
14 M. S. Kiraz, M. A. Bingöl, and S. Kardaş: Inter. J. Inf. Security Sci. **1** (2012) 32.
15 N. Li, Y. Mu, and W. Susilo: Radio Frequency Identification: Security and Privacy Issues (Springer, Berlin, 2014) 108.
16 Z. Ding, J. Li, and B. Feng: J. Comput. Res. Dev. **46** (2009) 583 (in Chinese).
17 S. Wang, S. Liu, and D. Chen: J. Comput. Res. Dev. **50** (2013) 1276 (in Chinese).
18 D. Boneh and M. Franklin: CRYPTO 2001 (Springer, Berlin, 2001) p. 213.

About the Authors

Jinbin Zheng graduated from Xiamen University in 2004. He is a college lecturer in the School of Mathematics and Information Engineering, Longyan University. His research interests are in the areas of information security, RFID system anonymity, and privacy protection technology.
(jbzheng518@163.com)

Fangguo Zhang is a professor in the School of Data and Computer Science, Sun Yat-sen University, and also a vice director of the Guangdong Provincial Key Laboratory of Information Security. His research interests are in the areas of cryptography theory and its applications. (isszhfg@mail.sysu.edu.cn)