A Lightweight NFC Authentication Algorithm Based on Modified Hash Function

Fang-Ming Cao[1] and Dao-Wei Liu[2]
(Corresponding author: Fang-Ming Cao)
School of Data and Computer Science, Guangdong Peizheng College[1]
Guangzhou 510800, China[1]
Email: caofangming1983@163.com
Department of Computer Science and Engineering, Guangzhou College of Technology and Business[2]
(Received Apr. 5, 2021; Revised and Accepted Apr. 5, 2022; First Online Apr. 23, 2022)

Abstract

In the NFC application, the leakage of user privacy information is inevitable. A lightweight NFC authentication algorithm based on a modified hash function is proposed to ensure the security of user privacy information. In the proposed algorithm, the specific implementation process using the modified hash function is given, which can provide better security requirements. In implementing the modified hash function, the Hamming weight variable of the encryption parameter itself is skillfully used, which reduces the introduction of parameters and improves the security factor of the encryption function. From the perspective of formalization and security analysis, it can be shown that the proposed algorithm satisfies the security needs of users. Furthermore, the simulation result demonstrates that the cost of the proposed algorithm is better than other NFC algorithms.

Keywords: Hamming Weight; Lightweight; Authentication Algorithm; Modified Hash Function; Near Field Communication (NFC); Privacy Leakage

1 Introduction

Near field communication (NFC) is a kind typical short-distance communication technology that can read out the information stored in objects without contacting themself [14]. The NFC technology is evolved from RFID technology, due to the low cost of the electronic tag in the RFID system, the computing power of the electronic tag is seriously restricted, it is unable to carry out the encryption and decryption calculation based on the traditional cryptography algorithm, resulting in the leakage of user privacy information [16, 17]. In order to improve the computing power of electronic tags and ensure the security of users' privacy information, therefore, the NFC technology is produced.

In general, the communication principle of NFC technology is roughly the same as that of RFID technology. The main difference [7,9] is that the user device stored information in NFC system is no longer a simple electronic tag device, but a mobile device with powerful computing power, such as mobile phone. The mobile phones have strong computing power and data storage capacity, and can carry out encryption and decryption calculation based on traditional cryptography algorithm, which can better protect the security of user privacy information. So, mobile phones can not only replace the traditional RFID electronic tags, but also bring great convenience to users. For example, users can generate electronic tag records, such as bus cards and bank cards on mobile phones, which can reduce the amount of things users bring when they go out [4, 8, 13].

This paper is organized as follows: In Section 1, we introduce the background of NFC technology. And then, we review the related work in Section 2. In Section 3, a lightweight NFC authentication algorithm based on modified hash function is proposed. In Section 4, the security analysis of NFC authentication algorithm is given. In Section 5, we analyzed the performance of NFC authentication algorithm. In Section 6, from the formal point of view, the NFC algorithm based on GNY logic formal reasoning is analyzed. In Section 7, simulation experiment is carried out from the perspective of energy consumption in the communication process. Lastly, we conclude the whole paper in Section 8.

2 Related Works

At present, NFC security research has attracted more and more attention. According to the analysis of NFC algorithm in [3], the third party can track the label position by observing the last round of failed sessions, which makes the algorithm unable to provide forward security. In [15], an authentication algorithm based on hash function is given. The disadvantage is that the tag side will call hash function for many times to calculate, which increases the calculation burden on the tag side. In [12], the

authors mainly analyzed the algorithm designed by CHO, and gave an improved algorithm, which uses hash function to complete encryption, and the computation cost at the tag side is proportional to the number of tags, which is not suitable for large-scale tag authentication environment.

In [11], an authentication algorithm based on hash function and elliptic curve is designed, which?s main drawback is that when hash function is called for more than ten times, elliptic curve needs to be called for nearly ten times. So, it can‘t be used in the existing system with the limited computation. In [2], Duc et al. proposed a new authentication algorithm?which is still based on hash function, and can‘t provide forward security and resist asynchronous attacks.

In [10], in order to resist tag location and tracking? an authentication algorithm is proposed, which has certain security requirements. But, the drawback is that the calculation cost at the tag side is heavy. In [6], an algorithm based on hash function is given. Due to low computational complexity? this algorithm can be applied to the existing system. However, the algorithm lacks the authentication of the tag to the reader, so that the attacker can launch a fake attack.

In [1], an algorithm based on physical unclonable functions (PUF) is given. The algorithm uses PUF encryption and hash function, which increases the computation cost on the label side. At the same time, the algorithm does not update the share key information after each interaction, which makes the algorithm unable to resist asynchronous attacks and provide forward security. In [5], a provably secure algorithm based on hash function is given. There are many issues to be discussed in the algorithm analysis, which can be seen in the detailed analysis in the next section.

In view of the lack of security in most classical algorithms, this paper designs an algorithm combined with modified hash function to improve the security of the algorithm. In the proposed algorithm, according to the Hamming weight of the parameters, it increases the amount of encryption parameters, and carries out different encryption methods.

3 Design of the NFC Authentication Algorithm

In [5], the proposed algorithm uses hash function and pseudo-random number function to encrypt the algorithm simultaneously, which leads to high computation cost. This algorithm has the following problems. First, what is the meaning of the symbol *psID*? There is no detailed explanation in [5]. Second, what is the meaning of symbols *symbol*? There is also no detailed explanation in [5]. Thirdly, the operation length of $r \parallel t$ is $2l$. According to the join operation rule, the value of the first bit l is r, and the value of the last bit l is t. Then, the pseudo-random number and timestamp privacy information can be analyzed, and more other privacy information can be analyzed by combining these information with other information. Fourthly, after the message $r \parallel t$ sent to the tag by the server, the tag does not verify the message sent by the server, but directly carries out the follow-up operation, which leads to the potential security risk of fake attack.

Motivated by the above algorithm framework, and the following improvements are given. Firstly, the design symbols in the algorithm are explained in detail. Second, the algorithm discards the connection operation, the attacker can‘t obtain the privacy information. Third, important information is encrypted before it is sent, so the attacker only gets ciphertext. Fourth, the receiver first verifies the source of the message. If and only if the verification is passed, the receiver will carry out the following steps, so as to avoid irrelevant operations. The fifth point is the deformation of the encryption function. There is only one encryption parameter for the traditional hash function. After the deformation, there are two encryption parameters for the hash function. Based on the Hamming weight of the two parameters, different parameters are selected for encryption to increase the difficulty of cracking.

3.1 NFC Algorithm Symbol Description

The NFC algorithm symbols are described as follows:

R: The whole constituted by the server and the reader, unified as the server;

T: Mobile devices, such as mobile phones;

K: The secret value shared between R and T;

K_{old}: The secret value shared between R and T in the last round;

K_{new}: The shared secret value between R and T in the current round;

T_{IDS}: T pseudonym;

T_{ID}: T identifier;

r_T: Random number generated by T;

r_R: Random number generated by R;

$\oplus$: Bitwise XOR operation;

&: Bitwise sum operation;

$h(X, Y)$: The modified hash function, X, Y are two encryption parameters. When the Hamming weight value X is large, it will encrypt Y; otherwise, it will encrypt X;

ASK: Start session command;

B, D, E, F, M: Communication message.

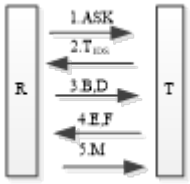

Figure 1: NFC Algorithm Steps

3.2 Steps of the NFC Authentication Algorithm

The proposed algorithm can be divided into two different stages [18], the first stage is the initialization stage, and the second stage is the authentication stage. The purpose of the first stage is to complete the initialization of the R and T information. After the initialization, the R end stores the information as $K, K_{old}, K_{new}, T_{IDS}, T_{ID}$, and $K_{old} = K_{new} = K$; the T end stores the information as K, T_{IDS}, T_{ID}.

The steps of proposed algorithm can be seen in Figure 1. The specific steps are described as follows.

Step 1. Starts the algorithm from the R end, and R sends a message ASK to T.

Step 2. After receiving the message, T sends its own pseudonym T_{IDS} to r as a response.

Explanation: T sends a pseudonym T_{IDS} without sending the real identifier information T_{ID}, which can ensure the security of T privacy information; at the same time, the pseudonym T_{IDS} will be updated after each communication, so as to avoid the attacker launching a location tracking attack on T.

Step 3. After R receiving the message, searches in the database to see if the data is equal to the received T_{IDS}? No, the algorithm stops. If there is, R will generate a random number r_R , and attain the message B, D according to the calculation rule, and finally send B, D to T.

Among them, $B = r_R \oplus T_{ID}, D = h(r_R, T_{ID})$.

Explanation: if R is not found in the database, it indicates that T is likely to be counterfeited by the attacker. Due to this step operation, the redundant steps can be avoided in the follow-up R. If found, R can retrieve other information related to the tag according to T_{IDS}, for example T_{ID}, to facilitate subsequent message calculation.

Step 4. After receiving the message, T will deform the received message B to get the random number r_R, and then use the same algorithm to calculate D by combing r_R with itself T_{ID}. At the same time, it will judge the relationship between D and D.

If the relation is unequal, the algorithm stops.

If the relation is equal, it means $r_R = r_R, D = D$, and it also means that R passes the verification of T. At this time, T generates a random number r_T, calculates the value of message E, F in turn according to the agreed rules, and finally sends message E, F to R. where message $E = (r_R \& T_{ID}) \oplus r_T, F = h(r_T \oplus K, K), D = h(r_R, T_{ID}) = h(B \oplus T_{ID}, T_{ID})$; random number $r_R = B \oplus T_{ID}$.

Explanation: the main function of the message B is to get the random number generated by R, and the main function of the message D is to judge the authenticity of R.

Step 5. After R receiving the message, R will deform the received message E to get the random number r_T, and then use the same algorithm to get F by combing r_T with itself K_*. At the same time, it will judge the relationship between F and F.

If the relation is unequal, the algorithm stops.

If the relation is equal, it indicates that $r_T = r_R$, and T is verified by R. At this point, R starts to calculate the message M, then updates the information, and finally sends M to T.

In the above, when $* = old$, R updates the information in the following way: $K_{new} = h(K_{old}, r_R \& r_T), T_{IDS}^{new} = h(T_{IDS}, r_R \& r_T)$.

In the above, when $* = new$, R updates the information in the following way:

$$\begin{aligned} K_{old} &= K_{new}, \\ K_{new} &= h(K_{new}, r_R \& r_T), \\ T_{IDS}^{new} &= h(T_{IDS}, r_R \& r_T). \end{aligned}$$

where message $M = h(r_R, r_T)$, $F = h(r_T \oplus K_{new}, K_{new})$ or $F = h(r_T \oplus K_{old}, K_{old})$; random number $r_T = E \oplus (r_R \& T_{ID})$.

Explanation: R use K_{new} to calculate F first, and only when the verification T fails, R will use K_{old} again to calculate another F and verify T again. The two verifications can resist the desynchronization attack initiated by the attacker.

Step 6. After T receiving the message, obtains M‘ by combining with the random number of the previous calculation according to the agreed algorithm, and judges the relationship M‘ with M.

If the relation is unequal, the algorithm stops.

If the relation is equal, it indicates that M‘ $= M$, and it also indicates that R passes verification by T. T start updating information:

$$\begin{aligned} K &= h(K, r_R \& r_T), \\ T_{IDS} &= h(T_{IDS}, r_R \& r_T). \end{aligned}$$

After T completes the information update, the algorithm ends.

Explanation: In the process of calculating messages M‘, we need to use a random number r_R, which has been calculated in Step 4, so it can be directly used here.

4 Security Analysis for NFC Algorithm Protocol

Mutual Authentication.
The proposed algorithm can provide authentication between the two parties in each communication. Specifically, in Step 3, R verifies T through information T_{IDS} for the first time; in Step 5, R verifies T through information E, F for the second time. The verification of R by T is realized in Step 4 and Step 6. Specifically, in Step 4, T verifies R through information B, D for the first time; in Step 6, T verifies R through information M for the second time. So, the proposed algorithm can provide mutual authentication.

Forward Security.
In the encryption process, random numbers are added into session message B, D, E, F, M. Random numbers are randomly generated by random number generator, which are mutually different and unpredictable. It is not feasible to analyze the random number used in the next session from the eavesdropping message for the attacker, so the attacker cannot pass the verification. Therefore, the algorithm has forward security.

Backward Security.
During the encryption process, the session messages B, D, E, M keep the fresh by random numbers r_R, while the session messages E, F, M keep the fresh by random numbers r_T. Random numbers are randomly generated, and the random numbers used in the previous two rounds of conversation have no correlation. The attacker can‘t deduce the previous random number from the current random number, so, the attacker cannot analyze the useful privacy information. Therefore, the proposed algorithm has backward security.

Replay Attack.
In order to pass a session entity verification?the attacker replays the message obtained from the previous round of eavesdropping in the next round session, so as to obtain other privacy information. The proposed algorithm adds random number to all message encryption process in order to solve the replay attack initiated by the attacker. When the attacker replays the message of the previous round, the random value used in the next round session has changed, so that the calculated value of the current round message is also different, and the replay of the previous round message can only fail to verify. Therefore, the algorithm can resist replay attack.

Location Attack.
In the proposed algorithm, the mobile device identifier information is hidden deliberately, and the pseudonym is introduced. Pseudonym can make the attacker unable to know the real mobile device identifier information. At the same time, pseudonym also uses the update mechanism after each information exchange, which makes the pseudonym information used in each round different, so that the attacker can‘t locate the specific location of the mobile device. Therefore, the proposed algorithm can resist location attack.

Fake Attack.
The premise of successful fake attack is that the attacker needs to have the privacy information owned by the real session device, and in the algorithm, no matter what means the attacker uses, he can‘t know the privacy information of any session device in advance; at the same time, in the communication, all the information is sent after encrypted, so that the attacker can eavesdrop on the message in plaintext and cannot know the privacy information. Therefore, the proposed algorithm can resist fake attack.

Asynchronous Attack.
The attacker can destroy some information of the server or mobile device in the session, so that the consistency between them will be lost and the normal communication will not be achieved. In the proposed algorithm, the share key of two rounds sessions is stored at one server end. In Step 5, the server will verify the two mobile devices, so that the consistency between the two can be restored. Therefore, the algorithm can resist asynchronous attack.

5 Performance Analysis of NFC Algorithm

In this section, the comparative analysis between the proposed algorithm and other classical algorithms is given from the calculation cost and the interaction number in mobile devices. The analysis results are shown in Table 1. Notes: $\surd$ means can resist, $\times$ means cannot resist.

In Table 1, the symbol P represents the bitwise operation (such as and operations, join operations, XOR operations), the symbol r represents the computation cost of generating random number, the symbol h represents the computation cost of hash functions (or modified hash functions), the symbol e represents the computation amount of elliptic curve encryption, and the symbol f represents the operation of physically unclonable functions. The symbol pr indicates the computation amount of pseudo-random number function. The symbol l represents the length of each session message, and the symbol

Table 1: Performance and Security Comparison of Different Algorithms

Attack type	[13]	[16]	[10]	[4]	[9]	**Our protocol**
Calculation amount	$6p+1r+8h$	$2p+2r+3h+4e$	$2p+1r+6h$	$1p+1r+4h+6f$	$1p+1r+4h+3pr$	$3p+1r+5h$
Traffic	$7l+1bit$	$6l+2bit$	$6l$	$8l+2bit$	$7l$	$6l+1bit$
Mutual authentication	√	√	√	√	×	√
Forward security	×	√	√	×	√	√
Backward security	√	√	√	√	√	√
Replay attack	√	√	√	√	√	√
Location attack	√	×	√	√	√	√
Fake attack	√	√	×	√	×	√
Asynchronous attack	×	√	√	×	√	√

Notes: √ means can resist, × means cannot resist.

bit represents a bit length. The symbol √ means that it can resist the attack; the symbol × means that it cannot resist the attack.

According to the analysis in Table 1, the computation cost of the proposed algorithm is similar to that of the algorithm in [10, 13] on the mobile device end, but the computation amount of the proposed algorithm is still slightly small on the mobile device end, because the number of encryption times of hash function is less than the two other algorithms. The proposed algorithm is different from the algorithm in [4,9,16] in terms of the computation amount at the mobile device end. Because the algorithm in [4, 9, 16] not only uses hash function encryption, but also uses other encryption methods to encrypt information, which increases the computation amount to a certain extent. At the same time, the total number of gates at the mobile device side will also increase in the implementation process, resulting in an increase of mobile devices. From the perspective of interactive information number in a complete session, the proposed algorithm basically maintains a considerable level compared with other algorithms.

In the proposed algorithm, bit operation will be used for the first time when calculating random number r_R, and bit operation will be used for the second and third time when calculating message E. Therefore, in the whole algorithm, bit operation is used for three times on the mobile device end. The mobile device side will generate a random number r_T, so in the whole algorithm, the random number generation is used on the mobile device side. The mobile device side uses hash function encryption for the first time when calculating messages D, and will use hash function encryption for the second time, the third time, the fourth time and the fifth time when calculating messages F, message M, updating shared key K and updating pseudonym T_{IDS}. Therefore, hash function encryption will be used on the mobile device side for five times in the whole algorithm. Based on the above, the total computation cost at the mobile device side is $3p+1r+5h$.

Based on the analysis of Table 1, the proposed algorithm can reduce the computation cost at the mobile device side. In a whole session, this algorithm does not increase the number of interactive information, meanwhile, the proposed algorithm has a greater improvement in the security aspect compared with other algorithms. It can give the common types attacks, provide the better security requirements to users, and ensure the security of users' privacy information.

6 Formal Reasoning Based on GNY Logic

The formal reasoning method based on GNY logic is used to prove the algorithm.

Formal Model.

In order to analyze the proposed algorithm formally with GNY logic, it is necessary to model the communication process in the proposed algorithm formally. Here, symbol R is used to represent server and symbol T is used to represent the mobile device:

$Msg1 : R \rightarrow T : ASK$

$Msg2 : T \rightarrow R : T_{IDS}$

$Msg3 : R \rightarrow T : B, D$

$Msg4 : T \rightarrow R : E, F$

$Msg5 : R \rightarrow T : M$

The model is further formulated as follows:

$Msg1 : T \triangleleft *ASK \sim | \rightarrow R| \equiv \#ASK$

$Msg2 : R \triangleleft *T_{IDS} \sim | \rightarrow T| \equiv \#T_{IDS}$

$Msg3 : T \triangleleft * (B, D) \sim | \rightarrow R| \equiv \# (B, D)$

$Msg4 : R \triangleleft * (E, F) \sim | \rightarrow T| \equiv \# (E, F)$

$Msg5 : T \triangleleft *M \sim | \rightarrow R| \equiv \#M$

Initialization Hypothesis.

$A1 : R \ni T_{ID}$

$A2: R \ni K$

$A3: R \ni T_{IDS}$

$A4: T \ni T_{IDS}$

$A5: T \ni T_{ID}$

$A6: T \ni K$

$A7: R| \equiv \#(r_T)$

$A8: T| \equiv \#(r_R)$

$A9: R| \equiv R \overset{T_{ID}}{\longleftrightarrow} T$

$A10: R| \equiv R \overset{T_{IDS}}{\longleftrightarrow} T$

$A11: R| \equiv R \overset{K}{\longleftrightarrow} T$

$A12: T| \equiv T \overset{K}{\longleftrightarrow} R$

$A13: T| \equiv T \overset{T_{ID}}{\longleftrightarrow} R$

$A14: T| \equiv T \overset{T_{IDS}}{\longleftrightarrow} R$

Initialization hypothesizes are as follows: $A1, A2, A3$ is owned by server R, $A4, A5, A6$ is owned by mobile device T, $A7$ is server R's belief in the freshness of information, $A8$ is server T's belief in the freshness of information, $A9, A10, A11$ is the share information trusted between the mobile device T and server R, $A12, A13, A14$ is the share information trusted between the server R and mobile device T.

Proving Goals.

Based on GNY logic formal analysis, in the proposed algorithm?five goals needs to be proved:

$G1: R| \equiv T| \sim \#(F)$

$G2: R| \equiv T| \sim \#(E)$

$G3: T| \equiv R| \sim \#(B)$

$G4: T| \equiv R| \sim \#(D)$

$G5: T| \equiv R| \sim \#(M)$

Reasoning Proof.

Because the reasoning proof process of the five proof goals is similar, here, only the proof of $G1$ is selected as an example for reasoning proof. The reasoning of proof goal is as follows:

First of all, from the initialization assumption $A7: R| \equiv \#(r_T)$: and the freshness rule: $\frac{P|\equiv\#(X)}{P|\equiv\#(X,Y),P|\equiv\#(F(X))}$ we can know that $R| \equiv \#(r_T, K)$.

In $Msg4, R \triangleleft * r_T$, that is $R \ni r_T$, combining the initialization assumptions $A1, A2, A3$, and rules $P2$, we can know that $R \ni (r_T, K)$.

Then, according to $R| \equiv \#(r_T, K)$, $R \ni (r_T, K)$ and the rule of freshness $F10: \frac{P|\equiv\#(X),PX}{P|\equiv\#(H(X,Y))}$, we can know that $R| \equiv \#(F)$, that is $R| \equiv \#(h(r_T \oplus K, K))$.

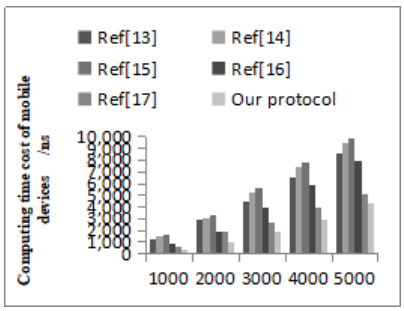

Figure 2: Comparison of Computing Time Cost of Mobile Devices with Different Algorithms

Finally, according to $Msg4, A11, R \ni (r_T, K), R| \equiv \#(h(r_T \oplus K, K))$ and message parsing rule $I3$, we can get: $R| \equiv T| \sim (F)$, that is $R| \equiv T| \sim (h(r_T \oplus K, K))$.

From the definition of freshness, we can deduce the proof goal $G1: R| \equiv T| \sim (F)$, that is $G1: R| \equiv T| \sim (h(r_T \oplus K, K))$.

7 Simulation Experiment

In this section, simulation experiments are carried out for the computing time cost of mobile devices in different NFC algorithms. The computing time cost of the mobile device not only includes the computing time at the mobile device side, but also includes the waiting time in the whole message interaction process. Therefore, the result is related to the computation amount and the communication amount.

The simulation environment is as follows: win 8 operating system (64 bit operating system), 4GB ram and Intel Core i5-3230m CPU@ 60GHz. Small and portable MySQL is used as the database for data storage, MATLAB software is used for simulation, part of the simulation program is based on C language programming, and the data storage is realized by linked list in data structure.

In the simulation experiment, in order to avoid the interference of random factors on the accuracy of simulation experiment, no less than 200 simulation experiments are carried out in each simulation experiment. There is only one server, and gradually increase the number of mobile devices that have a session with the server. When the number of mobile devices in the session is 1000, 2000, 3000, 4000 and 5000, we record each computing time cost at the mobile device side, calculate the average value of experiment data, and take the average value as the final result. The computing time cost of different algorithms at the mobile device side is shown in Figure 2.

As can be seen from Figure 2, when the number of mobile devices is small, the computing time overhead of the mobile devices is close among different algorithms. When the number of mobile devices interacting with the server

increases gradually, the computing time cost of the algorithms in [4, 9, 16] increases obviously. The main reason is that the three algorithms not only use hash function, but also use other encryption algorithms, which makes the overall computing time cost increase obviously. The computing time cost of mobile devices between [10, 13] and our algorithm is not very large. The main reason is that the three algorithms only use hash function to encrypt information, which can effectively reduce the computing time cost. However, it is obvious that the computing time cost of our algorithm is better than that of the other two algorithms. The reason is that the number of times using hash function encryption is slightly less than other algorithms. Therefore, overall, the total communication cost of the proposed algorithm is better than other algorithms.

8 Conclusions

In this paper, a lightweight NFC authentication algorithm is proposed. In the proposed algorithm, we firstly present a modified hash function encryption algorithm, which encrypts the communication information to ensure the information security. The modified hash function makes full use of the Hamming weight parameter carried by the encryption parameter itself, which can reduce the introduction of new parameters, decrease the storage space, meanwhile, increase the difficulty of attacker's cracking capacity. The analysis results show that the proposed algorithm has higher security performance and meets the user's security needs. Simulation experiment results show that the comprehensive cost of the proposed algorithm is better than other NFC algorithms.

Acknowledgments

This paper is supported by the Exploration on curriculum system reform of excellent network engineer training under the background of "collaborative education"(the second batch of industry university collaborative education project of Higher Education Department of the Ministry of education in 2018) (201802068001).

References

[1] X. Chen, K. Choi, K. Chae, "A secure and efficient key authentication using bilinear pairing for NFC mobile payment service," *Wireless Personal Communications*, vol. 92, no. 1, pp. 1–17, 2017.

[2] S. Y. Chiou, "An efficient RFID authentication protocol using dynamic identity," *International Journal of Network Security*, vol. 21, no. 5, pp. 728–734, 2019.

[3] Y. P. Duan, "Lightweight RFID group tag generation protocol," *Control Engineering of China*, vol. 27, no. 4, pp. 751–757, 2020.

[4] P. Gope, J. Lee, T. Q. S. Quek, "Lightweight and practical anonymous authentication protocol for RFID systems using physically unclonable functions," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2831–2843, 2018.

[5] Y. J. Li, S. B. Wang, X. T. Yang, *et al.*, "Lightweight NFC security authentication scheme based on mobile terminal," *Computer Engineering and Applications*, vol. 56, no. 16, pp. 84–89, 2020.

[6] J. Ling, Y. Wang, W. F. Chen, "An improved privacy protection security protocol based on NFC," *International Journal of Network Security*, vol. 19, no. 1, pp. 39–46, 2017.

[7] D. W. Liu, J. Ling, "Improved RFID authentication protocol with backward privacy," *Computer Science*, vol. 43, no. 8, pp. 128–130, 2016.

[8] Y. L. Liu, X. C. Yin, Y. Q. Dong, *et al.*, "Lightweight authentication scheme with inverse operation on passive RFID tags," *Journal of the Chinese Institute of Engineers*, vol. 42, no. 1, pp. 74–79, 2019.

[9] Z. H. Liu, C. J. Huang, H. Suo, "Modified mobile RFID bidirectional authentication protocol against counterfeiting attack," *Computer Applications and Software*, vol. 37, no. 6, pp. 309–315, 2020.

[10] Y. N. Ma, "NFC communications-based mutual authentication scheme for the internet of things," *International Journal of Network Security*, vol. 19, no. 4, pp. 631–638, 2017.

[11] S. Q. Mei, X. R. Deng, "Mobile RFID bidirectional authentication protocol based on shared private key and bitwise operation," *Computer Applications and Software*, vol. 37, no. 7, pp. 302–308, 2020.

[12] F. Tan, "An improved RFID mutual authentication security hardening protocol," *Control Engineering of China*, vol. 26, no. 4, pp. 783–789, 2019.

[13] J. Q. Wang, Y. F. Zhang, D. W. Liu, "Provable secure for the ultra-lightweight RFID tag ownership transfer protocol in the context of iot commerce," *International Journal of Network Security*, vol. 22, no. 1, pp. 12–23, 2020.

[14] P. Wang, Z. P. Zhou, J. Li, "Improved serverless RFID security authentication protocol," *Journal of Frontiers of Computer Science and Technology*, vol. 12, no. 7, pp. 1117–1125, 2018.

[15] Y. S. Wei, J. H. Chen, "Tripartite authentication protocol RFID/NFC based on ECC," *International Journal of Network Security*, vol. 22, no. 4, pp. 664–671, 2020.

[16] R. Xie, J. Ling, D. W. Liu, "Wireless key generation algorithm for RFID system based on bit operation," *International Journal of Network Security*, vol. 20, no. 5, pp. 938–949, 2018.

[17] H. Xu, J. Ding, P. Li, et al, "A lightweight RFID mutual authentication protocol based on physical unclonable function," *Sensors*, vol. 18, no. 3, pp. 760–780, 2018.

[18] B. Yuan, J. Liu, "A universally composable secure grouping proof protocol for RFID tags," *International Journal of Network Security*, vol. 28, no. 6, pp. 1872–1883, 2016.

Biography

Fang-ming Cao graduated from South China University of technology with a master's degree in 2016. Now working in Guangdong Peizheng University, he is a full-time teacher of computer major, and also a lecturer. At present, the research direction is mainly in the field of information security, etc.

Dao-wei Liu received a master's degree in School of Computers from Guangdong University of Technology (China) in June 2016. He is a teacher in department of computer science and engineering, Guangzhou College of Technology and Business. His current research interest fields include information security.