ECC-based RFID/NFC Mutual Authentication Protocol

Weiguang Huo^{1, a}, Qingkuan Dong^{2, b} and Yuan Chen^{3, c}

¹Xidian University, Xi'an, Shaanxi, 710071, China

²State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an, Shaanxi, 710071, China

³Xidian University, Xi'an, Shaanxi, 710071, China

^ahuoweiguang2010@126.com, ^bqkdong@xidian.edu.cn

Keywords: IoT RFID/NFC Authentication ECC Security Analysis

Abstract. RFID (Radio Frequency IDentification) is a generic term for technologies that use radio signals to automatically identify objects in the IoT (Internet of Things). NFC (Near Field Communication) originating from RFID is a wireless peer-to-peer communication technology, which is widely applied to mobile phones for contactless payment. The open link between the RFID tag and reader is vulnerable to various attacks, such as eavesdropping, tampering and impersonating, etc. PKC (Public Key Cryptography) based authentication is able to solve the above problem. Compared with authentication methods based on symmetric key, PKC is more secure, flexible and suitable for large-scale RFID/NFC applications due to its flexible key management and high security. However, the traditional PKC technologies are not able to be directly applied to low-cost RFID tags because of the limited computing and storage capacity of tags. To alleviate this issue, we propose a low-cost RFID authentication protocol based on ECC (Elliptic Curve Cryptography) that has high security level and small size parameters. The protocol combines the cryptoGPS identification protocol and the precomputing based Diffie-Hellman key exchange method which is proposed in this paper. The security of the protocol is analyzed and proved under the random oracle model.

Introduction

RFID is a generic term for technologies that use radio signals to automatically identify objects in the IoT. A typical RFID system is mainly composed of three parts: the tag, reader and back-end database [9]. The tag attached on the object has a unique electronic product code, which acts as the identity of the object. The reader is the device that connects tags to the back-end database. The back-end database is responsible for completing the integration, management and exchange of the data. RFID technology has been widely employed in public transportation, security access control, food safety, logistics tracking and medical management etc. Recently NFC which is derived from RFID as a wireless peer-to-peer communication technology in the IoT, has made a good figure in the electronic payment and smart media. Compared with an RFID system, the slight difference is that an NFC device must be able to be a reader as well as a tag, and thus we can treat an NFC device as a special RFID system. In an RFID system, the open link between the RFID tag and reader is vulnerable to many attacks, such as eavesdropping, tampering and impersonating, etc. Although NFC is intrinsically reliable for mobile payments [1,2] due to its short range, it also faces the same problem. Authentication can provide an efficient and reliable solution to security and privacy problems, therefore many authentication protocols based on symmetric key cryptography appear gradually. However, symmetric key based authentication is not suitable for large-scale RFID applications.

PKC based authentication is able to solve the above problem, which is more secure, flexible and suitable for large-scale RFID/NFC applications than symmetric key based authentication due to its key management and high security. But traditional PKC technologies are not able to be applied to low-cost RFID tags because of the limitations of the computing power and storage capacity of tags. In this paper, we propose a low-cost RFID mutual authentication protocol based on ECC that has high security level and small size parameters. Compared with the existing research, the proposed protocol

is designed by using improved precomputed Diffie-Hellman key exchange and cryptoGPS identification technology, and after the authentication a session key is also established for subsequent data transmission. Considering the circuit area, power consumption and computational capability of low-cost tags, the necessary results are stored in tag's memory by precomputing the scalar multiplication on elliptic curve which is complex and time-consuming. Then the authentication between the tag and reader becomes very easy. In addition, the back-end database and back-end search are not required in our protocol because a public key management method is used in the process of authentication. In this way, the back-end database has no need to store public keys of tags, which makes the RFID system have excellent scalability.

Related Work

As is well known, PKC is slower in speed, higher in computation complexity and consumes more resources than symmetric key cryptography, but ECC has greatly eased the embarrassment. Due to its small key lengths and high efficiency, ECC has become an important component of the design of RFID authentication protocols.

In 2008, Lee et al. [4] presented an ECC-based RFID authentication protocol, which claimed to offer anonymity and untraceability and reduce the computing load of tags, but in fact their intended target was not reached. From 2009 to 2011, Refs. [5,6,7,8] introduced their own idea by improving Lee et al.'s protocol. Unfortunately, these protocols were proved to be defective later.

In 2009, Ahamed et al. [9] proposed ERAP protocol, which could effectively prevent location tracking and tag forgery but could not resist DoS attack. In 2010, Lee et al. [10] put forward an ECC-based authentication protocol which only implemented one-way authentication. Batina et al. [11] raised a hierarchical ECC-based RFID authentication protocol in 2011, but the protocol could be subject to side channel attack. In 2013, Liao et al. [12] dropped out a new protocol which could satisfy security requirements for RFID systems. However, it would bring very heavy computational burden to the tag when the protocol ran. One recent protocol was presented by Moosavi et al. [14]. It claimed that their protocol had a higher security level, 48% less communication overhead and 24% less total memory than related work, but the tag still needed to calculate elliptic curve scalar multiplication in their protocol.

CryptoGPS identification protocol that was proposed by Girault, Poupard and Stern is a landmark. The use of precomputations, which we recommend here, is suitable for low-cost tags. In 2007, Mcloone and Robshaw [3] proposed cryptoGPS digital signature scheme, which was based on cryptoGPS identification protocol. This scheme involving only 160-bit elliptic curve scalar multiplication could make the tag less calculation and provide higher security. However, the scheme above only achieved one-way authentication from reader to tag, the search complexity of back-end database was high, and there was likely to be vulnerable to DoS attack for tags. In 2014, Dong et al. [13] improved and optimized cryptoGPS identification protocol. In their protocol, LHW technology and coupons-updating algorithm were introduced, with the result that it made up for disadvantages of cryptoGPS identification protocol.

Through the above introduction, it is obvious that many of the existing ECC-based RFID authentication protocol is not secure, what's more, tags need to calculate several elliptic curve scalar multiplications, which is impossible for the low-cost tag due to its limited computing power, storage space and chip area. However, the idea of using precomputation is feasible, which inspires us to design a new protocol.

Security Requirements and Security Model

In this section, some basic security requirements suitable for any occasion will be described. A security model to meet the security requirements above is established, and it is mainly composed of two parts: the ability of the adversary and the security definition.

Security Requirements. When designing a protocol, security requirements of an RFID system are widely concerned. The basic security requirements include mutual authentication, confidentiality, anonymity, availability and forward security, and they are defined as follows.

Mutual authentication: Two communication entities must identify each other by authentication. For RFID systems, authentication protocol helps the tag and reader to know whether the other side is trustworthy or not.

Confidentiality: In the process of data exchange between the tag and reader, the information transmitted in the channel should not be accessed by illegal users. Thus, data usually needs to be encrypted, which ensures that only the legitimate receiver can access the information.

Anonymity: Anonymity can prevent the adversary from obtaining location information of the object by the tag attached to it, which achieves the purpose of protecting user's privacy.

Availability: In readiness for correct services, an excellent protocol should be able to resist common attacks, and any malicious adversary cannot affect the operational capabilities of the system.

Forward security: In the operation of the protocol, an adversary can eavesdrop the current communication content, but he cannot trace the previously transmitted information using the present.

Security Model. BR security model [15] is proposed by Bellare and Rogaway for proving the security of the session key distribution protocol under the provable security framework. In this model there are three types of entities: that is tag T , reader R and the PPT (Probabilistic Polynomial Time) adversary A . The model is described as follows:

Adversary model: We denote the authentication protocol as P , and each operation of the protocol is called an instance of P . Because the authentication between the tag and reader is likely to be more than once in different application scenarios, there may be many protocol instances simultaneously. We denote the i -th protocol instance of tag T as Π_T^i , and the j -th protocol instance of reader R as Π_R^j . Each instance of the protocol can be modeled as a PPT oracle. We assume that an adversary A can enquire the oracle, whose ability is embodied in the different kinds of queries. Adversary A is allowed to execute any of the following queries:

Execute(Π_T^i, Π_R^j): Adversary A can eavesdrop the detailed communication process of the protocol instances Π_T^i and Π_R^j by this query, and the output is all of the communication information from an honest instance of protocol P .

Send(Π_T^i, Π_R^j, m): Adversary A sends message m to the protocol instances Π_T^i and Π_R^j . Π_T^i and Π_R^j process message m according to protocol P , and then return the results to adversary A .

Reveal(Π_T^i): This query is used to confirm whether the two session keys are independent of each other.

Corrupt(T): Adversary A can bribe the tag by this query, then tag T takes the initiative in disclosing the secret information stored in its memory.

Test(Π_T^i, Π_R^j): The purpose of this query is to measure the semantic security of the secret session key from protocol instances Π_T^i and Π_R^j .

Security definition: In the experiment of attack, adversary A can conduct the first four queries polynomial times. These queries formalize various attacks that adversary A may carry out, such as eavesdropping, replay, bribing a legitimate entity, etc. When he finishes enquiring, adversary A needs to guess the returned value of the *Test* session and identify whether the value is the true secret session key or a random number. Adversary A will output a b' after receiving this value. If $b=b'$, adversary A do the experiment successfully. Thus adversary A tries to enquire at most $q(n)$ times within polynomial time $t(n)$, his advantage Adv in the experiment is:

$$Adv = \Pr[Exp_A(q, t)] - 1/2 = \Pr[b=b'] - 1/2, \quad (1)$$

where Exp is the experiment done by adversary A .

The Proposed Protocol

In this section, we will introduce the proposed ECC-based RFID mutual authentication protocol, which can satisfy the basic security requirements in the application of RFID system. The tag can conduct the authentication process efficiently by using simple arithmetic, which reduces its computational burden. In addition, we put forward an injection protocol for the tag to update the precomputed values stored in tag's memory if these values are exhausted.

The proposed protocol mainly consists of three phases: initialization, authentication, and injection of precomputed values. Prior to illustrating the process of the protocol, we assume that the communication channel between the reader and back-end database is secure, and the channel between the tag and reader is insecure. The protocol we proposed aims to solve the potential security problems between the tag and reader and to provide reliable communication environment for them. The notations used in the protocol are explained in Table 1.

Table 1. Notations and their descriptions

Notations	Descriptions
$GF(q)$	finite fields $GF(q)$, and q is prime
P	elliptic curve base point over $GF(q)$
ID_R	reader's identity
s_T, s_R	tag and reader's private key
Q_T, Q_R	tag and reader's public key
$CERT$	tag's digital certificate
k_{ij}	session key restored in the tag
(r_v, U_v)	precomputed values
(λ, A)	precomputed values used for the TTP (Trusted Third Party)
Q_S	public key for the TTP
θ	Session key between the tag and the TTP
$H()$	hash function
$E_k(), D_k()$	encryption and decryption algorithm with key k
$\parallel$	concatenation operator
$\oplus$	XOR

Initialization Phase. In the initialization phase, some precomputed values need to be injected into the tag's memory, and this process is done by the TTP. The precomputed values are shown in Table 2.

Table 2. Precomputed values

	ID_{R1}	ID_{R2}	...	ID_{Rn}
$r_1, U_1=r_1P$	k_{11}	k_{12}	...	k_{1n}
$r_2, U_2=r_2P$	k_{21}	k_{22}	...	k_{2n}
...	...	...	...	...
$r_l, U_l=r_lP$	k_{l1}	k_{l2}	...	k_{ln}

$r_1, r_2, \dots, r_l$ are random numbers, and $k_{ij}=H(r_j(Q_{Rj}))$ is the precomputed session key, $1 \leq i \leq l, 1 \leq j \leq n$, Q_{Rj} is the reader R_j 's public key obtained by $s_{Rj}P$ (s_{Rj} is the reader R_j 's privacy key), and both $s_{Rj}P$ and $r_j(Q_{Rj})$ are elliptic curve scalar multiplications. The advantage of the method above is that a tag can support n readers to access its memory and each reader has l chances of authentication. After that, the TTP still needs to prestore the random number λ , precomputed value $A=\lambda P$ and session key $\theta=H(\lambda(Q_S))$ (it has the same structure as k_{ij}) in the tag. These data will be used for authentication between the TTP and tag before the TTP starts to inject the precomputed values into the tag.

Authentication Phase. As is shown in Fig. 1, the authentication process between the tag and reader is as follows:

Step 1: The reader chooses a random number r_z , and sends $ID_{Rz}||r_z$ to the tag;

Step 2: After receiving the query message $ID_{Rz}||r_z$, the tag looks up the reader's identity ID_{Rz} in Table 2. If it is not found, the tag will refuse the reader to access, otherwise it will proceed as follows: Randomly choose a pair of data (r_v, U_v) in Table 2, and read k_{vz} relevant to ID_{Rz} and (r_v, U_v) , then

choose a random number r' and calculate $C=E_{k_{vz}}(CERT||r_z||r')$, finally send $C||U_v$ to response to the reader;

Step 3: After receiving the tag's response $C||U_v$, the reader calculates $K=H(S_{Rz}(U_v))=k_{vz}$, decrypts C with key K and checks whether r_z and $CERT$ are valid. If both of them are valid, the reader freely chooses a random number c and calculates $C'=K \oplus (c||r')$, then sends it to the tag;

Step 4: The tag gets c and r' after receiving C' . If r' is not identical to what the reader sent before, the authentication will be terminated, otherwise the tag calculates $y=r_v+cS_T$ and $C''=K \oplus y$, then sends C'' to the reader. Finally the used key $K=k_{vz}$ is reset to 0;

Step 5: The reader obtains y from C'' and computes U_v+cQ_T , then it verifies the tag by checking whether $yP=U_v+cQ_T$. If $yP=U_v+cQ_T$, the mutual authentication between the tag and reader is accomplished, and K as a session key is used for subsequent communication.

Fig. 1. ECC-based RFID mutual authentication protocol.

Precomputed Values Injection Phase. If precomputed values are exhausted in the tag for one reader, the TTP can inject new values by the following method:

Step 1: The TTP runs the above authentication protocol with the tag using the precomputed values λ , A and θ , if they authenticate each other, go to step 2;

Step 2: The TTP updates all the used precomputed values in the tag, together with λ , A and θ .

Security Analysis and Performance Comparison

In this section we will analyze the security of the proposed protocol. We will also compare the performance of our protocol with some recent ones.

Security Analysis. Using the security model defined, to prove that the proposed protocol is secure is to prove the following theorem:

Theorem 1: Under the security model, if adversary A has finished the experiment and his advantage $Adv=\Pr[Exp_A^h(q,t)]-1/2$ is a negligible value, then the protocol is secure.

Proof: In the ideal simulation environment, we use a random function g as an oracle for adversary enquiring. In this case, if adversary A experiments successfully, the output is 1, then let $\Pr[Exp_A^g(q,t)=1]$ be its probability.

In the primary stage of the attack, adversary A needs to disguise himself as a reader R and transmit the challenge to a tag T , and then obtain essential knowledge set for attack. After that, adversary A sets out impersonating the tag T and using reader's challenge in the knowledge set to guess the true value of b :

(1) If the challenge has been enquired by adversary A , he has the ability to guess that either $ID_{Rz}||r_{z1}$ or $ID_{Rz}||r_{z0}$ participates in the authentication. After conducting $\{Execute, Send, Reveal, Corrupt\}$ queries at most $q(n)$ times, adversary A must guess the result by using *Test* query, the probability is $(q(n)+1)/2^n$.

(2) If the challenge has not been enquired by adversary A before, he cannot be clear all the possible output of random function g as an oracle, and the probability that he guesses successfully is $1/2$.

In summary, all this leads up to the successful probability of the experiment in the ideal simulation environment:

$$\Pr[Exp_A^g(q,t)=1] \leq (q(n)+1)/2^n + 1/2. \quad (2)$$

In order to determine the successful probability of experiment in the real environment, we will replace the random function g with hash function h to simulate the oracle. If the simulation of the oracle is random function g , let I be this event, then the successful probability of event I is the same as the experiment done by adversary A :

$$\Pr[I(g)=1] = \Pr[Exp_A^g(q,t)=1]. \quad (3)$$

When the oracle calls hash function h to answer the equeries from adversary A , we have

$$\Pr[I(h)=1] = \Pr[Exp_A^h(q,t)=1]. \quad (4)$$

Then we can get

$$\Pr[I(h)=1] - \Pr[I(g)=1]. \quad (5)$$

Let $\varepsilon(n) = \Pr[Exp_A^h(q,t)=1] - 1/2$, we have

$$\Pr[Exp_A^h(q,t)=1] - \Pr[Exp_A^g(q,t)=1] \geq \varepsilon(n) - (q(n)+1)/2^n. \quad (6)$$

The simulation of the oracle may call the real random function or the hash function, which is indistinguishable in polynomial time, so we can get

$$\varepsilon(n) \leq (q(n)+1)/2^n + \delta(n), \quad (7)$$

where $\delta(n)$ is a tiny value. If n is very large, $\varepsilon(n)$ is negligible, and we prove that the proposed protocol is secure in the random oracle model. In the BR security model, our protocol with semantic security can resist known session key attack. In the following, we demonstrate the security analysis in details according to the security requirements that our protocol needs to meet.

Mutual Authentication: As aforementioned, we adopt a new key management method in the protocol. Namely in the initialization phase, the public key of the legitimate reader is prewritten into the tag's memory directly by the TTP, which helps the tag to verify the reader initially by the public key when the tag receives the reader's request. Since the response message that the tag sends to the reader is encrypted, only the legitimate reader is able to extract the real information. Moreover, benefited from the employment of the session key $k_{vz} = H(S_{Rz}(U_v))$ that is similar to the Diffie-Hellman key exchange and the introduction of hash function, the man-in-the-middle attack can be averted. That is to say, the authentication from the tag to the reader in Step 4 is protected. The reader needs to verify the legitimacy of the tag's certificate *CERT* in Step 3 and the equation $yP = U_v + cQ_T$ in Step 5 to finish authenticating the tag.

Confidentiality: The proposed protocol also has an effect of the key agreement, and thus key can be assured that the subsequent communication is secure. Restricted by the tag storage capacity, the number of the keys prestored in the tag is limited, but it will be updated with the depletion of

prestored keys. Actually, the range of possible values for the keys is much wider. Besides, the random number contained in the session key makes each authentication use different keys.

Forward Security: In the authentication phase, an adversary can eavesdrop all the messages because of the open communication channel. In our protocol, the random number in each message is employed to provide freshness and different structures for the message, as a result, the adversary cannot track the previous session by the current. Even if the adversary steals the session key and know the information about certificate further, he cannot track the previous session because the random number contained in the session key is private.

Privacy/Traceability Resistance: As the random numbers that tag and reader select, the prestored value (r_v, U_v) , and the session key k_{vz} are different, the messages transmitted in the channel will be different during each authentication process. The adversary cannot track the tag because he does not know the tag's identity information. Moreover, all information about the tag is encrypted, which can protect the data and location privacy.

Replay Attack Resistance: If an adversary impersonates a legal tag to conduct replay attack, he must interact with a reader with the information between the tag and reader that he has recorded previously. Because of the use of random numbers which are changed every time, it is very difficult to construct an available message. That is to say, the random numbers can prevent replay attack.

Impersonation Attack Resistance: At the beginning of the authentication, it is possible that an adversary impersonates a reader to defraud the tag of the trust. However, the impersonation is invalid since the adversary is not able to obtain the secret key of the reader. In order to avoid the impersonation of a tag, the authentication message is encrypted when the tag sends its certificate *CERT* to the reader. Hence the adversary is not able to get the certificate to impersonate the tag.

DoS Attack Resistance: An adversary might disguise himself as a legitimate reader in the process of authentication, so he is able to unremittingly send challenge to the tag to run out of the session keys prestored in the tag. However, the attack is impossible because the key space is much larger, although the number of the session keys prestored in the tag is limited. The key space will be updated and expanded with the depletion of prestored keys, so the protocol can resist DoS attack.

According to the above security analysis, we claim that the proposed protocol is suitable for the practical application of RFID systems. It guarantees the normal communication between the tag and reader, and improves the reliability of the RFID technology. Table 3 shows the security property comparison of the proposed protocol with recent researches.

Table 3. Security properties comparison

	M	C	F	P	R	I	D	N	S
Zhang et al. [8]	×	√	√	√	√	√	-	×	×
Lee et al. [10]	×	√	√	×	√	√	-	×	×
Liao et al. [12]	√	√	√	√	√	√	-	×	×
Moosavi et al. [14]	√	√	√	√	√	√	-	√	×
Ours	√	√	√	√	√	√	√	√	√

Note: √: Satisfied; ×: Not satisfied; -: Not applicable; M: Mutual authentication; C: Confidentiality; F: Forward security; P: Privacy; R: Replay attack resistance; I: Impersonation attack resistance; D: DoS attack resistance; N: No back-end database search; S: Scalability.

As illustrated in Table 3, the protocols proposed in [8,10] only realize one-way authentication from the reader to the tag, and thus incomplete authentication may lead the tag's memory to be read by an adversary because the tag cannot verify the legitimacy of the reader. In addition, privacy related to benefit is of great importance in RFID systems, and most protocols in Table 3 are capable of privacy protection. The protocols in [8,10,12,14] don't take into account the DoS attack, and our protocol can solve this problem very well. Furthermore, our protocol does not need back-end database in the process of authentication, which avoids the time-consuming back-end database search and makes the reader's position unrestricted no longer. The advantage without back-end database is that the RFID system has excellent scalability which effectively relieves the storage burden of the system due to the increase in the number of tags.

Performance Comparison. To illustrate the advantage of the proposed protocol further, in this section we will analyze the storage requirement, computational cost and hardware area on the tag. Besides the proposed protocol, several existing protocols are also compared with.

Storage Requirement: To adapt to the characteristic of RFID systems, we choose standardized 163-bit elliptic curve domain parameters recommended by NIST (National Institute of Standard and Technology). The proposed protocol needs to store the legitimate reader's identity information ID_R (and its public key Q_R), prestored value $(r, U=rP)$ and the related random session key $k=H(s_T(Q_R))$. These three values form the two-dimensional table in Table 2. The size of the table requires adaptive adjustment according to the storage capacity of the tag. The tag also needs to store its own certificate $CERT$ and public key s_T in order to provide authentication information for readers. In the protocol, the tag's identity information adopts 96-bit electronic product coding of EPC Global RFID standard, and the output of the hash function is 160 bits in order to reduce the risk of birthday attack. According to the montgomery algorithm [16] on standard projective coordinates, the input of the algorithm is only its affine x -coordinate, which can decrease the size of the messages.

Computational Cost: The computational cost of the tag is very low in our protocol, and the tag is able to complete the authentication process by using only one random number generation, one encryption and one 160-bit multiplication. Although the protocol is based on ECC which leads to a heavy computational burden, it can work with precomputation in the tag instead of calculating time-consuming scalar multiplications.

Table 4. Performance comparison

	Zhang et al. [8]	Lee et al. [10]	Liao et al. [12]	Moosavi et al. [14]	Ours
Storage requirement/bit	652	652	652	489	1250
Computational cost/S _m	3	3	5	3	0

Note: S_m denotes the number of elliptic curve scalar multiplications.

Table 4 shows the number of elliptic curve scalar multiplications which is time-consuming and resource-consuming compared with other simple operations. The proposed protocol requires more storage space of the tag. This can be explained by the fact that some values prestored in the tag because of the precomputation. However, this method can reduce the tag's computational burden effectively, which is benefit for low-cost tags. With the development of technology, the storage capacity of a tag will increase gradually, and the authentication protocol based on the precomputation will have a good future.

Summary

Considering the characteristics of the RFID system and its special security requirements, an ECC-based authentication protocol without back-end database is proposed in this paper. This protocol is designed by using improved precomputed Diffie-Hellman key exchange and cryptoGPS identification protocol, and realizes mutual authentication between the tag and reader. Benefited from the adoption of precomputation, the proposed protocol reduces computational cost and the resource consumption on the tag. In addition, a session key is negotiated at the end of the authentication, which can ensure the communication security. Finally, the protocol security analysis is conducted by using the provable security theory and method, and it proves that the protocol is secure in the random oracle model. Considering the advantages of cloud storage and the importance of the reader's anonymity, we will realize the authentication based on PKC among the tag, reader and cloud without back-end database for low-cost systems in the future.

Acknowledgements

This work is supported by National Natural Science Foundation of China (No. 61373172, 61402353).

Literature References

- [1] Pourghomi P, Ghinea G. Managing NFC payment applications through cloud computing[C]//Internet Technology And Secured Transactions, 2012 International Conference for. IEEE, 2012: 772-777.
- [2] Mareli M, Rimer S, Paul B S, et al. Experimental evaluation of NFC reliability between an RFID tag and a smartphone[C]//AFRICON, 2013. IEEE, 2013: 1-5.
- [3] M. McLoone and M.J.B. Robshaw, Public Key Cryptography and RFID Tags, M. Abe (Ed.):CT-RSA 2007, LNCS 4377, pp. 372–384, Springer, Heidelberg, 2007.
- [4] Lee, Y.K., Batina, L., Verbauwhede, EC-RAC (ECDLP based randomized access control): Provably secure RFID authentication protocol. RFID, pp. 97–104. IEEE, Los Alamitos, 2008
- [5] Lee Y K, Batina L, Verbauwhede I. Untraceable RFID authentication protocols: Revision of EC-RAC[C]//RFID, 2009 IEEE International Conference on. IEEE, 2009: 178-185.
- [6] Lee Y K, Batina L, Singelée D, et al. Low-cost untraceable authentication protocols for RFID[C]//Proceedings of the third ACM conference on Wireless network security. ACM, 2010: 55-64.
- [7] Lee Y K, Batina L, Singelée D, et al. Wide-Weak Privacy-Preserving RFID Authentication Protocols[M]//Mobile Lightweight Wireless Systems. Springer Berlin Heidelberg, 2010: 254-267.
- [8] Zhang X, Li J, Wu Y, et al. An ECDLP-Based Randomized Key RFID Authentication Protocol[C]//Network Computing and Information Security (NCIS), 2011 International Conference on. IEEE, 2011, 2: 146-149.
- [9] Ahamed S I, Rahman F, Hoque E. ERAP: ECC based RFID authentication protocol[C]//Future Trends of Distributed Computing Systems, 2008. FTDCS'08. 12th IEEE International Workshop on. IEEE, 2008: 219-225.
- [10] Y. Lee, L. Batina, D. Singele, B. Preneel, and I. Verbauwhede. Anti-counterfeiting, Untraceability and Other Security Challenges for RFID. In Towards Hardware-Intrinsic Security, pages 237–257. Springer, 2010.
- [11] Batina, L., Seys, S., Singelée, D., Verbauwhede, I.: Hierarchical ECC-Based RFID Authentication Protocol. In RFIDSec(2011)183-201
- [12] Liao Y P, Hsiao C M. A secure ECC-based RFID authentication scheme integrated with ID-verifier transfer protocol [J]. Ad Hoc Networks, 2014, 18: 133-146.
- [13] Dong Q, Ding W, Wei L. Improvement and optimized implementation of cryptoGPS protocol for low - cost radio - frequency identification authentication [J]. Security and Communication Networks, 2014.
- [14] Moosavi S R, Nigussie E, Virtanen S, et al. An Elliptic Curve-based Mutual Authentication Scheme for RFID Implant Systems [J]. Procedia Computer Science, 2014, 32: 198-206.
- [15] Wang Y. Efficient identity-based and authenticated key agreement protocol [M]//Transactions on Computational Science XVII. Springer Berlin Heidelberg, 2013: 172-197.
- [16] Karaklajic D, Fan J, Schmidt J M, et al. Low-cost fault detection method for ECC using Montgomery Powering Ladder[C]//Design, Automation & Test in Europe Conference & Exhibition (DATE), 2011. IEEE, 2011: 1-6.