Three elliptic curve cryptography-based RFID authentication protocols for Internet of Things

Rui An 1, Hui Feng 1 , Qin Liu 2, Li Li 3

1 School of Mathematics and Statistics, Wuhan University, Wuhan, China

ruia.whu@qq.com hfeng.math@whu.edu.cn

2 School of Computer, Wuhan University, Wuhan, China

csqliu@qq.com

3 International School of Software, Wuhan University, Wuhan, China

lli@whu.edu.cn

Abstract: With the development of information technology, the Internet of Thing (IoT) is extensively employed in many fields such as logistics, medical healthcare, food safety and intelligent transportation. The Radio Frequency Identification (RFID) technology is an important building block of the IoT. Therefore, how to address security problem in RFID system is a crucial issue for the security of the IoT. The RFID authentication protocol is a key cryptographic protocol ensuring communication security because it could provide authentication between the tag and the server. Recently, elliptic curve cryptography (ECC)-based RFID authentication protocols were studied widely because they could provide better security attributes compared with traditional RFID authentications. Lv et al. proposed three ECC-based RFID protocols and claimed their protocols could overcome weaknesses in previous protocols. Unfortunately, in this paper, we show that Lv et al.'s protocols cannot withstand the man-in-the-middle attack. To solve security problems in their protocols, we propose three improved ECC-based RFID authentication protocols.

Key words: Radio-frequency identification; Authentication protocol; Elliptic curve cryptography; Man-in-the-middle attack;

1. Introduction

The Internet of Things (IoT) is an emerging paradigm based on the modern wireless communication technology. Using embedded intelligence, the IoT could provide interconnections among different things including physical objects, cyber objects, and social objects [1]. With the development of many related technologies,

L. Barolli et al. (eds.), *Advances on Broad-Band Wireless Computing, Communication and Applications*, Lecture Notes on Data Engineering and Communications Technologies 2, DOI 10.1007/978-3-319-49106-6_88

such as communication technology, electrical production technology and system integration technology, the IoT has been extensively used in many fields including logistic management, supply chain management, electronic commerce, electronic government and industrial manufacturing. According to a recent study [2], about 50 to 100 billion things will be connected to the Internet through the IoT by 2020. Due to wireless communication, the IoT is more vulnerable to different attacks compared with the traditional networks. Therefore, how to solve the security problem in the Iot become a very important issue in practical applications.

To expand the application of the IoT, many technologies and network devices such as the Radio Frequency Identification (RFID), wireless sensor networks and cloud computation have been used in the IoT. As an important building block of the IoT, the RFID technology attracted worldwide attentions from different fields. As an important automatic identification and data capture technology, the RFID technology is introduced during the Second World War. It could identify different objects such as goods and animal using radio waves. Compared with the traditional barcode technology, the RFID technology has many advantages: 1). Providing both read capability and write capability; 2). Providing the function of reading many tags synchronously; 3). Requiring no line-of-sight contact. Therefore, it could be applied in many environments and considered as the best replacement of the traditional barcode technology. According to a recently study [3], the market value of the RFID technology will gross over USD 25 billion in 2018.

RFID authentication protocol is an important security protocol for ensuring secure communication in RFID systems because it could provide authentication between the tag and the server. Due to the limited computing power and storage of the tag, it is difficult to design authentication protocols for RFID systems. Many RFID authentication protocols [4-13] using XOR operations or hash function operations or pseudo-random number generator have been proposed. According to Lee et al.'s study [14], the Elliptic Curve Cryptography (ECC) is also suitable for the design of RFID authentication protocol. Several ECC-based RFID authentication protocols [14-17] have been proposed to support mutual authentication between the tag and the server. The authentication process of those protocols is very complicated. In many applications such as logistic management and supply chain management, only the function that the server could authenticate the tag is needed. Compared with ECC-

based RFID authentication protocols supporting mutual authentication, ECC-based RFID authentication protocols supporting single authentication have better performance.

Lee et al. [18] proposed an Elliptic Curve Discrete Logarithm (ECDL) problem based randomized access control (EC-RAC) protocols for single authentication in RFID systems. They demonstrated that their protocols were provably secure in the generic group model. Unfortunately, Bringer et al. [19] and Deursen et al. [20] pointed out that Lee et al.'s EC-RAC authentication protocols cannot withstand tracking attacks and replay attacks. To solve those security problems, Lee et al. [21] proposed three improved EC-RAC protocols. However, Deursen and Radomirovic [22] pointed out that Lee et al. improved EC-RAC protocols were still vulnerable to the tracking attacks. Lv et al. [23] also pointed out Lee et al.' protocols [21] were vulnerable to tracking attacks. To withstand tracking attacks, Lv et al. proposed three improved EC-RAC protocols. In this paper, we analyze the security of Lv et al.'s EC-RAC protocols. We demonstrate that their protocols cannot withstand the man-in-the-middle attacks. Afterwards, we proposed three improved EC-RAC protocols by modifying Lv et al.'s protocols slightly.

The organization of the paper is sketched as follows. Section 2 reviews Lv et al.'s EC-RAC protocols briefly. Section 3 analyzes the security of Lv et al.'s EC-RAC protocols. Section 4 proposes the improved EC-RAC protocols to solve problems in Lv et al.'s protocols. Security analysis and performance analysis are proposed in Section 5 and Section 6 respectively. At last, Section 7 gives some conclusions of the paper.

2. Review of Lv et al.'s protocols

To enhance security, Lv et al. proposed three ECC-based RFID authentication protocols, i.e. Lv et al.'s EC-RAC 1 protocol, Lv et al.'s EC-RAC 2 protocol and Lv et al.'s EC-RAC 3 protocol. For convenience, some notations used in the paper are defined as follows.

- $F(q)$: a finite filed;
- n: a large prime number;
- $E(F(q))$: an elliptic curve defined in $F(q)$;

- P: a point on $E(F(p))$ with order ;
- G: the group generated by the point P;
- (y, Y): the p n rivate/public key pair of the server, where $Y = yP$;
- (x_i, X_i): the secret information of the tag, where $X_i = x_i P, i = 1, 2$;

2.1. Lv et al.'s EC-RAC 1 protocol

This protocol is a kind of secure identity transfer scheme. In the protocol, the server could authenticate the tag by checking whether the received identity verifier is stored in its database. At the beginning, the server chooses system parameters $params = \{F(q), E(F(q)), n, P, Y\}$. It also stores (X_1) and (x_1, Y) in its database and the tag's memory separately. As shown in Fig. 1, the following steps will be executed between the tag and the server.

1). The tag generates a random number r_{t1}, computes $T_1 = r_{t1}P$ and sends the message $\{T_1\}$ the server.

2). Upon receiving the message $\{T_1\}$, the server generates a random number r_{s1}, and sends the message $\{r_{s1}\}$ to the tag.

3). Upon receiving the message $\{r_{s1}\}$, the tag computes $T_2 = (r_{t1} + r_{s1}x_1 r_{t1})Y$ and sends the message $\{T_2\}$ to the server.

4). Upon receiving the message $\{T_2\}$, the server computes $U = r_{s1}^{-1}(y^{-1}T_2 - T_1)$. The server checks whether U and $x_1 T_1$ are equal. If they are not equal, the server rejects the session; otherwise, the tag is authenticated.

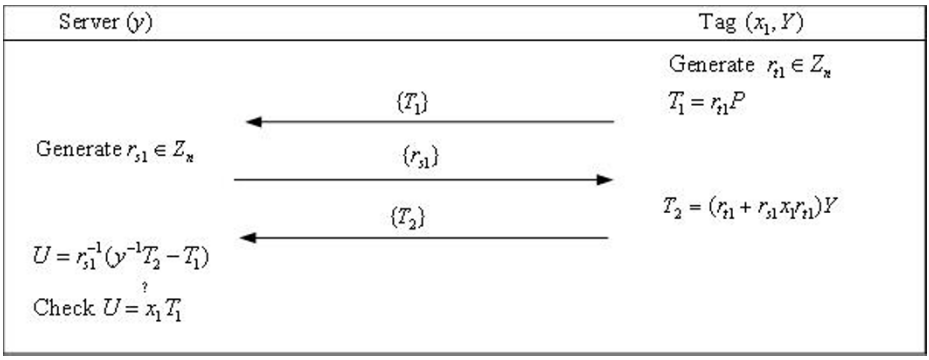

Fig. 1. Lv et al.'s Modified EC-RAC 1 protocol

2.2. Lv et al.'s EC-RAC 2 protocol

This protocol is a kind of secure identity transfer scheme and secure password transfer scheme. In the protocol, the server could authenticate the tag by checking whether the received identity verifier is stored in its database and the corresponding password is correct. At the beginning, the server chooses system parameters $params = \{F(q), E(F(q)), n, P, Y\}$. It also stores (x_1, X_1, x_2, X_2) and (x_1, x_2, Y) in its database and the tag's memory separately. As shown in Fig. 2, the following steps will be executed between the tag and the server.

1). The tag generates a random number r_{t1}, computes $T_1 = r_{t1}P$ and sends the message $\{T_1\}$ the server.

2). Upon receiving the message $\{T_1\}$, the server generates a random number r_{s1}, and sends the message $\{r_{s1}\}$ to the tag.

3). Upon receiving the message $\{r_{s1}\}$, the tag computes $T_2 = (r_{t1} + r_{s1}x_1r_{t1})Y$, $T_3 = (r_{t1}x_1 + r_{s1}x_2r_{t1})Y$ and sends the message $\{T_2, T_3\}$ to the server.

4). Upon receiving the message $\{T_2, T_3\}$, the server computes $W = r_{s1}^{-1}(y^{-1}T_2 - T_1)$ and $V = r_{s1}^{-1}(y^{-1}T_3 - x_1T_1)$. The server checks whether both equations $W = x_1T_1$ and $V = x_2T_1$ hold. If either of them does not hold, the server stops the session; otherwise, the tag is authenticated.

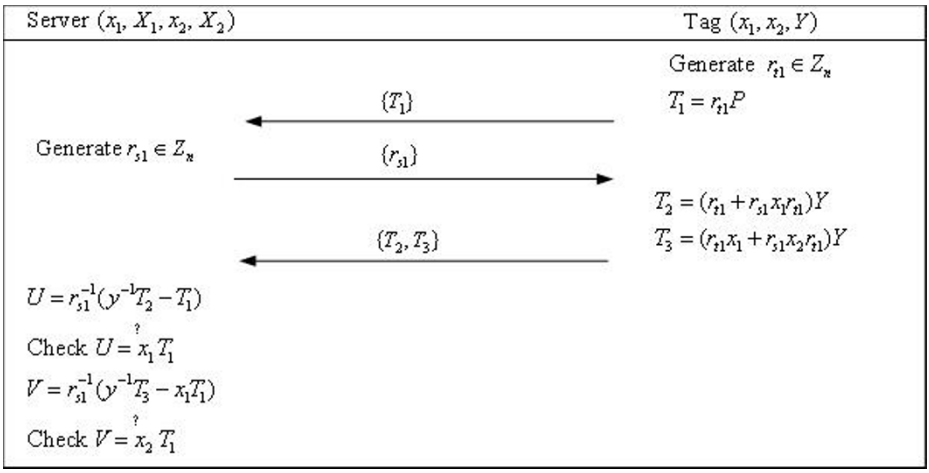

Fig. 2. Lv et al.'s Modified EC-RAC 2 protocol

2.3. Lv et al.'s EC-RAC 3 protocol

This protocol is a kind of secure identity transfer scheme and secure password transfer scheme. In the protocol, the server could authenticate the tag by checking whether the received identity verifier is stored in its database and the corresponding password is correct. At the beginning, the server chooses system parameters $params = \{F(q), E(F(q)), n, P, Y\}$. It also stores (x_1, X_1, x_2, X_2) and (x_1, x_2, Y) in its database and the tag's memory separately. As shown in Fig. 2, the following steps will be executed between the tag and the server.

1). The tag generates two random numbers r_{t1}, r_{t2}, computes $T_1 = r_{t1}P$, $T_2 = r_{t2}P$ and sends the message $\{T_1, T_2\}$ the server.

2). Upon receiving the message $\{T_1, T_2\}$, the server generates a random number r_{s1}, and sends the message $\{r_{s1}\}$ to the tag.

3). Upon receiving the message $\{r_{s1}\}$, the tag computes $T_3 = (r_{t1} + r_{s1}x_1r_{t1})Y$, $T_4 = (r_{t2}x_1 + r_{s1}x_2r_{t2})Y$ and sends the message $\{T_3, T_4\}$ to the server.

4). Upon receiving the message $\{T_3, T_4\}$, the server computes $U = r_{s1}^{-1}(y^{-1}T_3 - T_1)$ and $V = r_{s1}^{-1}(y^{-1}T_4 - x_1T_2)$. The server checks whether both equations $U = x_1T_1$ and $V = x_2T_2$ hold. If either of them does not hold, the server stops the session; otherwise, the tag is authenticated.

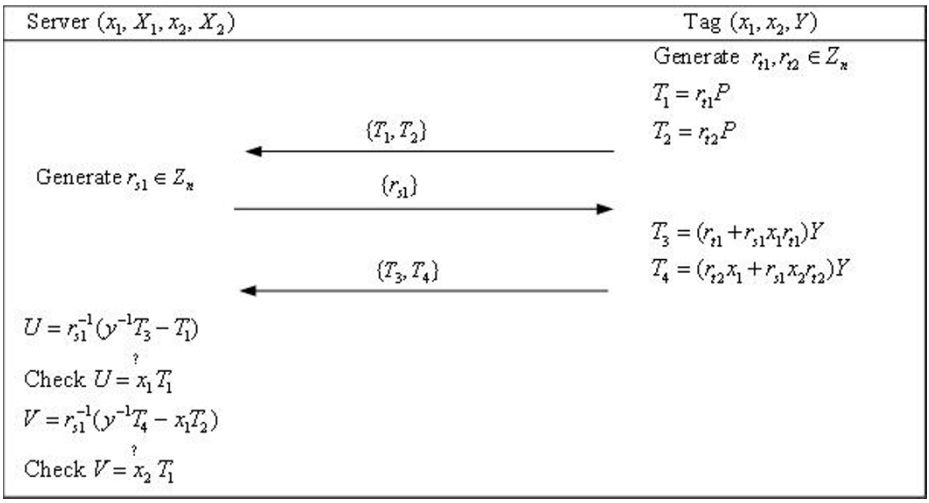

Fig. 3. Lv et al.'s Modified EC-RAC 3 protocol

3. Security analysis of Lv et al.'s protocols

With the development of the cryptographic theory, several security model for RFID authentication protocols have been proposed . According to Vaudenay's work, attackers against the RFID authentication protocols could be divided into wide (or narrow) attackers and strong (or weak) attackers. A wide (narrow) attacker is the one who could (not) get the verification result of the server. A strong (weak) attacker is the one who could (not) extract a tag's secret and reuse it. It is easy to say the wide–strong attacker is the most powerful. We call a RFID authentication protocol is wide-strong privacy-preserving if it is untraceable against the wide–strong attacker.

Lv et al. claimed that all their three protocols are wide-strong privacy-preserving against the wide–strong attacker. Unfortunately, we will show their protocols are not secure against the wide–strong attacker through proposing three concrete attacks.

3.1. Security analysis of Lv et al.'s EC-RAC 1 protocol

In this subsection, we analyze the security of Lv et al.'s EC-RAC 1 protocol. As show in Fig. 4, the man-in-the middle attack is described as follows.

1). The tag generates a random number r_{t1}, computes $T_1 = r_{t1}P$ and sends the message $\{T_1\}$ the server.

2). Upon intercepting the message $\{T_1\}$, the adversary generates a random number r_a, computes $T_1' = r_a T_1$ and sends message $\{T_1'\}$ to the server.

3). Upon receiving the message $\{T_1'\}$, the server generates a random number r_{s1}, and sends the message $\{r_{s1}\}$ to the adversary.

4). Upon receiving the message $\{r_{s1}\}$, the adversary sends it to the tag directly.

5). Upon receiving the message $\{r_{s1}\}$, the tag computes $T_2 = (r_{t1} + r_{s1}x_1 r_{t1})Y$ and sends the message $\{T_2\}$ to the server.

6). Upon intercepting the message $\{T_2\}$, the adversary generates a random number r_a, computes $T_2' = r_a T_2$ and sends message $\{T_2'\}$ to the server.

7). Upon receiving the message $\{T_2'\}$, the server computes $U = r_{s1}^{-1}(y^{-1}T_2' - T_1')$. The server checks whether U and $x_1 T_1'$ are equal. If they are not equal, the server rejects the session; otherwise, the tag is authenticated.

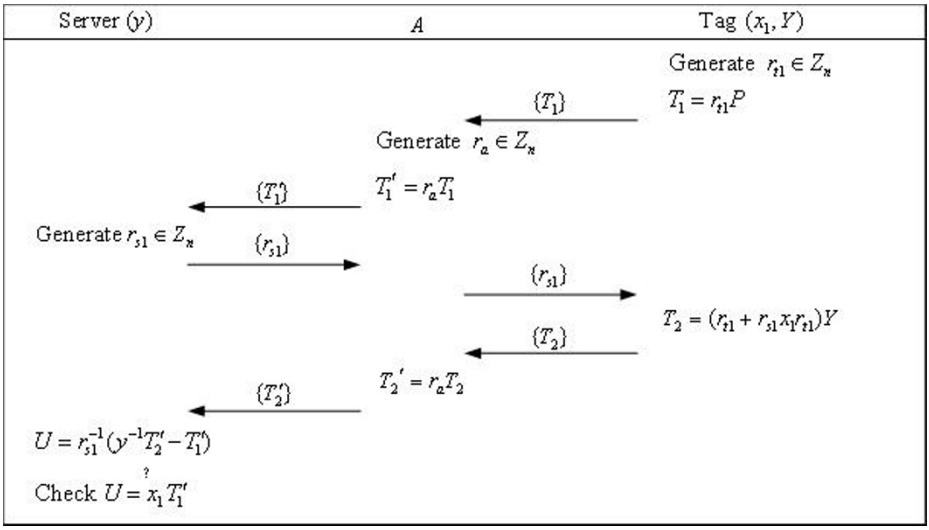

Fig. 4. Attack against Lv et al.'s Modified EC-RAC 1 protocol

Since $T_1 = r_{t1}P$, $T_1' = r_a T_1$, $T_2 = (r_{t1} + r_{s1}x_1 r_{t1})Y$ and $T_2' = r_a T_2$, then we could get that

$$\begin{aligned} U &= r_{s1}^{-1}(y^{-1}T_2' - T_1') = r_{s1}^{-1}(y^{-1}r_a T_2 - r_a T_1) \\ &= r_{s1}^{-1}(y^{-1}r_a(r_{t1} + r_{s1}x_1 r_{t1})Y - r_a r_{t1}P) \\ &= r_{s1}^{-1}(y^{-1}r_a(r_{t1} + r_{s1}x_1 r_{t1})yP - r_a r_{t1}P) \\ &= r_{s1}^{-1}(r_a(r_{t1} + r_{s1}x_1 r_{t1})P - r_a r_{t1}P) \\ &= r_{s1}^{-1}(r_a r_{t1}P + r_a r_{s1}x_1 r_{t1}P - r_a r_{t1}P) \\ &= r_{s1}^{-1}r_a r_{s1}x_1 r_{t1}P = x_1 r_a r_{t1}P = x_1 T_1' \end{aligned} \quad (1)$$

Thus, the message $\{T_1'\}$ and $\{T_2'\}$ could pass the verification of the server. Therefore, we can conclude that Lv et al.'s EC-RAC 1 protocol cannot withstand the man-in-the-middle attack.

3.2. Security analysis of Lv et al.'s EC-RAC 2 protocol

In this subsection, we analyze the security of Lv et al.'s EC-RAC 2 protocol. As show in Fig. 5, the man-in-the middle attack is described as follows.

1). The tag generates a random number r_{t1}, computes $T_1 = r_{t1}P$ and sends the message $\{T_1\}$ the server.

2). Upon intercepting the message $\{T_1\}$, the adversary generates a random number r_a, computes $T_1' = r_a T_1$ and sends message $\{T_1'\}$ to the server.

3). Upon receiving the message $\{T_1'\}$, the server generates a random number r_{s1}, and sends the message $\{r_{s1}\}$ to the adversary.

4). Upon receiving the message $\{r_{s1}\}$, the adversary sends it to the server directly.

5). Upon receiving the message $\{r_{s1}\}$, the tag computes $T_2 = (r_{t1} + r_{s1}x_1 r_{t1})Y$, $T_3 = (r_{t1}x_1 + r_{s1}x_2 r_{t1})Y$ and sends the message $\{T_2, T_3\}$ to the server.

6). Upon intercepting the message $\{T_2, T_3\}$, the adversary generates a random number r_a, computes $T_2' = r_a T_2$, $T_3' = r_a T_3$ and sends message $\{T_2', T_3'\}$ to the server.

7). Upon receiving the message $\{T_2', T_3'\}$, the server computes $W = r_{s1}^{-1}(y^{-1}T_2' - T_1')$ and $V = r_{s1}^{-1}(y^{-1}T_3' - x_1 T_1')$. The server checks whether both equations $W = x_1 T_1'$ and $V = x_2 T_1'$ hold. If either of them does not hold, the server stops the session; otherwise, the tag is authenticated.

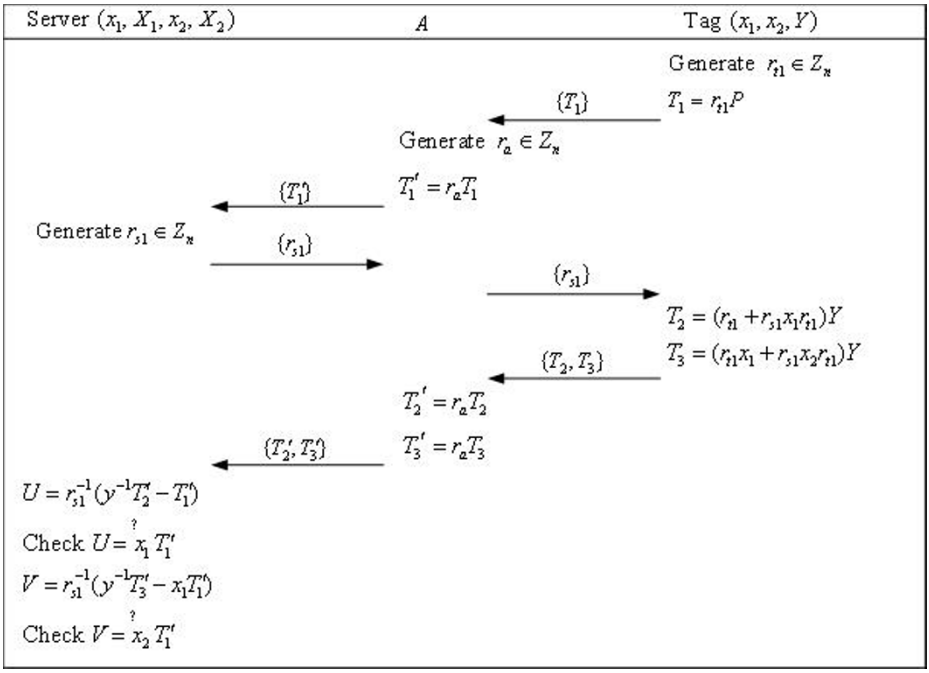

Fig. 5. Attack against Lv et al.'s Modified EC-RAC 2 protocol

Since $T_1 = r_{t1}P$, $T_1' = r_a T_1$, $T_2 = (r_{t1} + r_{s1}x_1 r_{t1})Y$, $T_3 = (r_{t1}x_1 + r_{s1}x_2 r_{t1})Y$, $T_2' = r_a T_2$ and $T_3' = r_a T_3$, then we could get that

$$\begin{aligned} U &= r_{s1}^{-1}(y^{-1}T_2' - T_1') = r_{s1}^{-1}(y^{-1}r_aT_2 - r_aT_1) \\ &= r_{s1}^{-1}(y^{-1}r_a(r_{t1} + r_{s1}x_1r_{t1})Y - r_ar_{t1}P) \\ &= r_{s1}^{-1}(y^{-1}r_a(r_{t1} + r_{s1}x_1r_{t1})yP - r_ar_{t1}P) \\ &= r_{s1}^{-1}(r_a(r_{t1} + r_{s1}x_1r_{t1})P - r_ar_{t1}P) \\ &= r_{s1}^{-1}(r_ar_{t1}P + r_ar_{s1}x_1r_{t1}P - r_ar_{t1}P) \\ &= r_{s1}^{-1}r_ar_{s1}x_1r_{t1}P = x_1r_ar_{t1}P = x_1T_1' \end{aligned} \tag{2}$$

and

$$\begin{aligned} V &= r_{s1}^{-1}(y^{-1}T_3' - x_1T_1') = r_{s1}^{-1}(y^{-1}r_aT_3 - x_1r_aT_1) \\ &= r_{s1}^{-1}(y^{-1}r_a(r_{t1}x_1 + r_{s1}x_2r_{t1})Y - x_1r_ar_{t1}P) \\ &= r_{s1}^{-1}(y^{-1}r_a(r_{t1}x_1 + r_{s1}x_2r_{t1})yP - x_1r_ar_{t1}P) \\ &= r_{s1}^{-1}(r_a(r_{t1}x_1 + r_{s1}x_2r_{t1})P - x_1r_ar_{t1}P) \\ &= r_{s1}^{-1}(r_ar_{t1}x_1P + r_{s1}x_2r_{t1}P - x_1r_ar_{t1}P) \\ &= r_{s1}^{-1}r_{s1}x_2r_{t1}P = x_2r_{t1}P = x_2T_1' \end{aligned} \tag{3}$$

Thus, the message $\{T_1'\}$ and $\{T_2', T_3'\}$ could pass the verification of the server. Therefore, we can conclude that Lv et al.'s EC-RAC 2 protocol cannot withstand the man-in-the-middle attack.

3.3. Security analysis of Lv et al.'s EC-RAC 3 protocol

In this subsection, we analyze the security of Lv et al.'s EC-RAC 2 protocol. As show in Fig. 5, the man-in-the middle attack is described as follows.

1). The tag generates two random numbers r_{t1}, r_{t2}, computes $T_1 = r_{t1}P$, $T_2 = r_{t2}P$ and sends the message $\{T_1, T_2\}$ the server.

2). Upon intercepting the message $\{T_1, T_2\}$, the adversary generates a random number r_a, computes $T_1' = r_aT_1$, $T_2' = r_aT_2$ and sends message $\{T_1', T_2'\}$ to the server.

3). Upon receiving the message $\{T_1', T_2'\}$, the server generates a random number r_{s1}, and sends the message $\{r_{s1}\}$ to the tag.

4). Upon receiving the message $\{r_{s1}\}$, the adversary sends it to the server directly.

5). Upon receiving the message $\{r_{s1}\}$, the tag computes $T_3 = (r_{t1} + r_{s1}x_1r_{t1})Y$, $T_4 = (r_{t2}x_1 + r_{s1}x_2r_{t2})Y$ and sends the message $\{T_3, T_4\}$ to the server.

6). Upon intercepting the message $\{T_3, T_4\}$, the adversary generates a random number r_a, computes $T_3' = r_aT_3$, $T_4' = r_aT_4$ and sends message $\{T_3', T_4'\}$ to the server.

7). Upon receiving the message $\{T_3', T_4'\}$, the server computes $U = r_{s1}^{-1}(y^{-1}T_3' - T_1')$ and $V = r_{s1}^{-1}(y^{-1}T_4' - x_1T_2')$. The server checks whether both equations $U = x_1T_1'$ and $V = x_2T_2'$ hold. If either of them does not hold, the server stops the session; otherwise, the tag is authenticated.

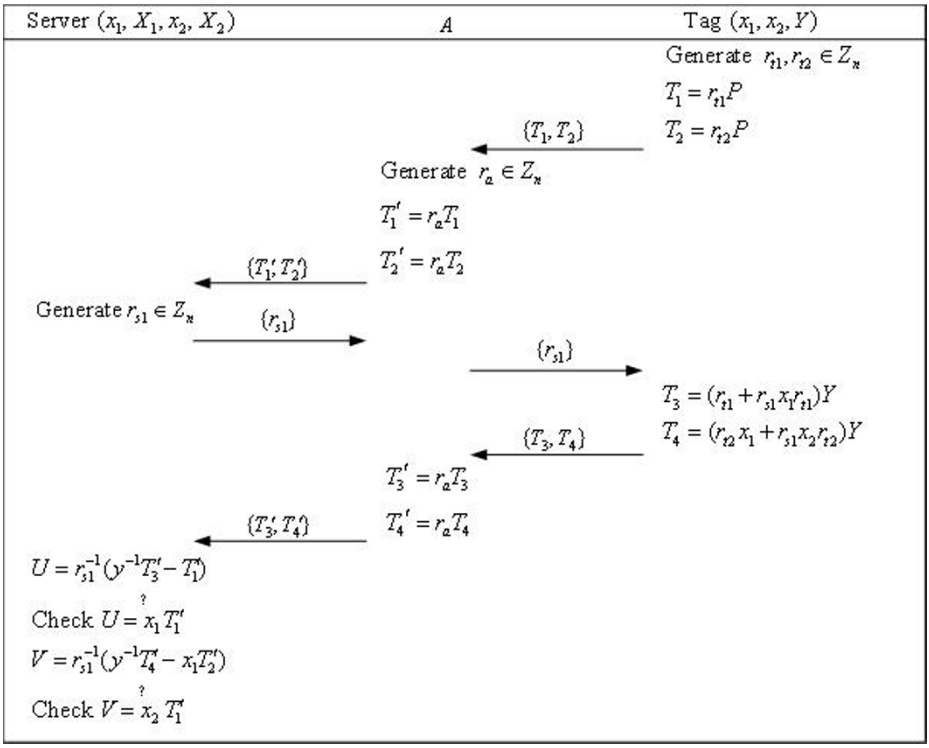

Fig. 6. Attack against Lv et al.'s Modified EC-RAC 3 protocol

Since $T_1 = r_{t1}P$, $T_2 = r_{t2}P$, $T_1' = r_a T_1$, $T_2' = r_a T_2$, $T_3 = (r_{t1} + r_{s1}x_1 r_{t1})Y$, $T_4 = (r_{t2}x_1 + r_{s1}x_2 r_{t2})Y$, $T_3' = r_a T_3$ and $T_4' = r_a T_4$, then we could get that

$$\begin{aligned} U &= r_{s1}^{-1}(y^{-1}T_3' - T_1') = r_{s1}^{-1}(y^{-1}r_a T_3 - r_a T_1) \\ &= r_{s1}^{-1}(y^{-1}r_a(r_{t1} + r_{s1}x_1 r_{t1})Y - r_a r_{t1}P) \\ &= r_{s1}^{-1}(y^{-1}r_a(r_{t1} + r_{s1}x_1 r_{t1})yP - r_a r_{t1}P) \\ &= r_{s1}^{-1}(r_a(r_{t1} + r_{s1}x_1 r_{t1})P - r_a r_{t1}P) \\ &= r_{s1}^{-1}(r_a r_{t1}P + r_a r_{s1}x_1 r_{t1}P - r_a r_{t1}P) \\ &= r_{s1}^{-1}r_a r_{s1}x_1 r_{t1}P = x_1 r_a r_{t1}P = x_1 T_1' \end{aligned} \tag{2}$$

and

$$\begin{aligned} V &= r_{s1}^{-1}(y^{-1}T_4' - x_1 T_2') = r_{s1}^{-1}(y^{-1}r_a T_4 - x_1 r_a T_2) \\ &= r_{s1}^{-1}(y^{-1}r_a(r_{t2}x_1 + r_{s1}x_2 r_{t2})Y - x_1 r_a r_{t2}P) \\ &= r_{s1}^{-1}(y^{-1}r_a(r_{t2}x_1 + r_{s1}x_2 r_{t2})yP - x_1 r_a r_{t2}P) \\ &= r_{s1}^{-1}(r_a(r_{t2}x_1 + r_{s1}x_2 r_{t2})P - x_1 r_a r_{t2}P) \\ &= r_{s1}^{-1}(r_a r_{t2}x_1 P + r_{s1}x_2 r_{t2}P - x_1 r_a r_{t2}P) \\ &= r_{s1}^{-1}r_{s1}x_2 r_{t2}P = x_2 r_{t2}P = x_2 T_2' \end{aligned} \tag{3}$$

Thus, the message $\{T_1', T_2'\}$ and $\{T_3', T_4'\}$ could pass the verification of the server. Therefore, we can conclude that Lv et al.'s EC-RAC 3 protocol cannot withstand the man-in-the-middle attack.

4. The proposed protocols

From the description of Lv et al.'s protocols, we know that there is linear relation between two messages sent by the tag. The linear relation could be used by the adversary to carry out the man-in-the-middle attacks. Subsequently, breaking the linear relation is the simplest way to withstand those attacks. Based on such thought, our protocols are described as follows.

4.1. Our EC-RAC 1 protocol

This protocol is a kind of secure identity transfer scheme. In the protocol, the server could authenticate the tag by checking whether the received identity verifier is

stored in its database. At the beginning, the server chooses system parameters $params = \{F(q), E(F(q)), n, P, Y\}$. It also stores (X_1) and (x_1, Y) in its database and the tag's memory separately. As shown in Fig. 7, the following steps will be executed between the tag and the server.

1). The tag generates a random number r_{t1}, computes $T_1 = r_{t1}P$ and sends the message $\{T_1\}$ the server.

2). Upon receiving the message $\{T_1\}$, the server generates a random number r_{s1}, and sends the message $\{r_{s1}\}$ to the tag.

3). Upon receiving the message $\{r_{s1}\}$, the tag computes $T_2 = (r_{t1}x(T_1) + r_{s1}x_1r_{t1})Y$ and sends the message $\{T_2\}$ to the server, where $x(T_1)$ denotes the x-coordinate of the elliptic curve point T_1.

4). Upon receiving the message $\{T_2\}$, the server computes $U = r_{s1}^{-1}(y^{-1}T_2 - x(T_1)T_1)$. The server checks whether U and x_1T_1 are equal. If they are not equal, the server rejects the session; otherwise, the tag is authenticated.

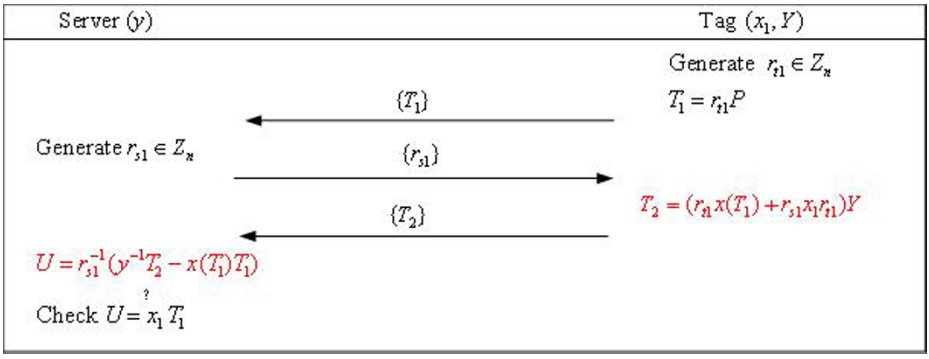

Fig. 7. Lv et al.'s Modified EC-RAC 1 protocol

4.2. Our EC-RAC 2 protocol

This protocol is a kind of secure identity transfer scheme and secure password transfer scheme. In the protocol, the server could authenticate the tag by checking whether the received identity verifier is stored in its database and the corresponding

password is correct. At the beginning, the server chooses system parameters $params = \{F(q), E(F(q)), n, P, Y\}$. It also stores (x_1, X_1, x_2, X_2) and (x_1, x_2, Y) in its database and the tag's memory separately. As shown in Fig. 8, the following steps will be executed between the tag and the server.

1). The tag generates a random number r_{t1}, computes $T_1 = r_{t1}P$ and sends the message $\{T_1\}$ the server.

2). Upon receiving the message $\{T_1\}$, the server generates a random number r_{s1}, and sends the message $\{r_{s1}\}$ to the tag.

3). Upon receiving the message $\{r_{s1}\}$, the tag computes $T_2 = (r_{t1}x(T_1) + r_{s1}x_1r_{t1})Y$, $T_3 = (r_{t1}x_1x(T_1) + r_{s1}x_2r_{t1})Y$ and sends the message $\{T_2, T_3\}$ to the server, where $x(T_1)$ denotes the x-coordinate of the elliptic curve point T_1.

4). Upon receiving the message $\{T_2, T_3\}$, the server computes $W = r_{s1}^{-1}(y^{-1}T_2 - x(T_1)T_1)$ and $V = r_{s1}^{-1}(y^{-1}T_3 - x_1x(T_1)T_1)$. The server checks whether both equations $W = x_1T_1$ and $V = x_2T_1$ hold. If either of them does not hold, the server stops the session; otherwise, the tag is authenticated.

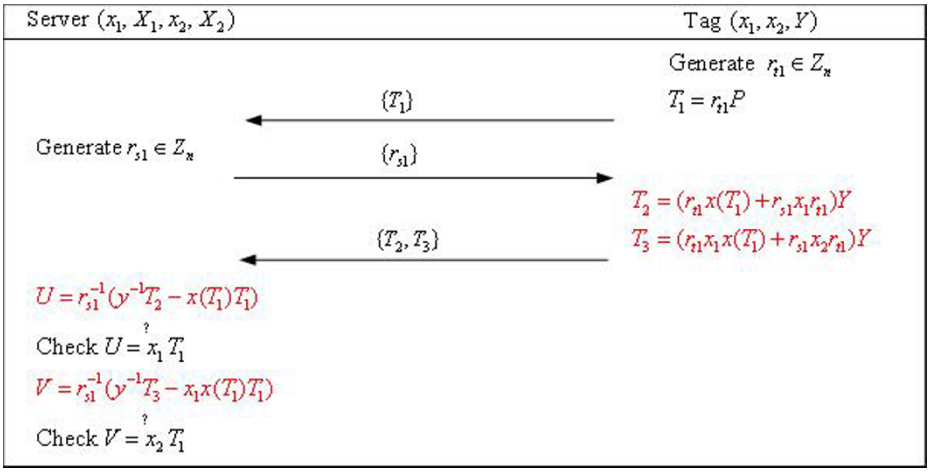

Fig. 8. Lv et al.'s Modified EC-RAC 2 protocol

4.3. Our EC-RAC 3 protocol

This protocol is a kind of secure identity transfer scheme and secure password transfer scheme. In the protocol, the server could authenticate the tag by checking whether the received identity verifier is stored in its database and the corresponding password is correct. At the beginning, the server chooses system parameters $params = \{F(q), E(F(q)), n, P, Y\}$. It also stores (x_1, X_1, x_2, X_2) and (x_1, x_2, Y) in its database and the tag's memory separately. As shown in Fig. 9, the following steps will be executed between the tag and the server.

1). The tag generates two random numbers r_{t1}, r_{t2}, computes $T_1 = r_{t1}P$, $T_2 = r_{t2}P$ and sends the message $\{T_1, T_2\}$ the server.

2). Upon receiving the message $\{T_1, T_2\}$, the server generates a random number r_{s1}, and sends the message $\{r_{s1}\}$ to the tag.

3). Upon receiving the message $\{r_{s1}\}$, the tag computes $T_3 = (r_{t1}x(T_1) + r_{s1}x_1r_{t1})Y$, $T_4 = (r_{t2}x_1x(T_2) + r_{s1}x_2r_{t2})Y$ and sends the message $\{T_3, T_4\}$ to the server, where $x(T_1)$ and $x(T_2)$ denote the x-coordinate of the elliptic curve points T_1 and T_2 respectively.

4). Upon receiving the message $\{T_3, T_4\}$, the server computes $U = r_{s1}^{-1}(y^{-1}T_3 - x(T_1)T_1)$ and $V = r_{s1}^{-1}(y^{-1}T_4 - x_1x(T_2)T_2)$. The server checks whether both equations $U = x_1T_1$ and $V = x_2T_2$ hold. If either of them does not hold, the server stops the session; otherwise, the tag is authenticated.

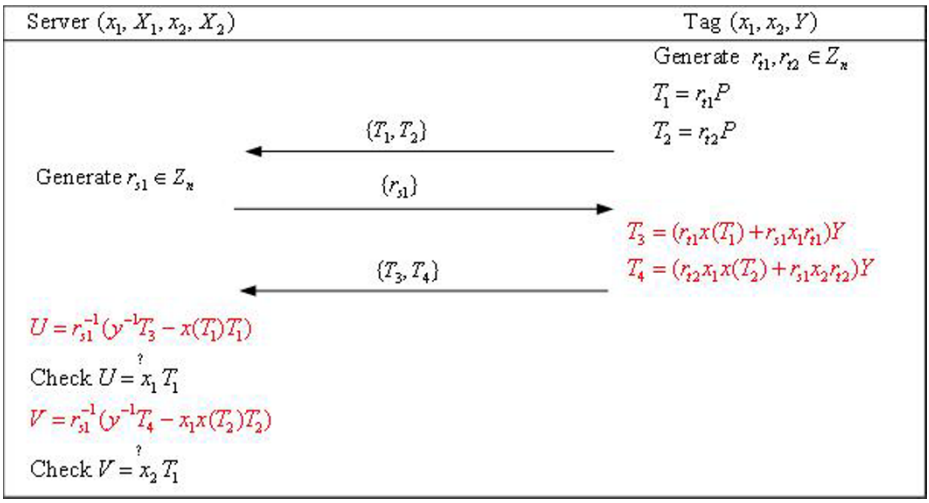

Fig. 9. Lv et al.'s Modified EC-RAC 3 protocol

5. Security analysis

In this section, we just analyze the security of our EC-RAC 1 protocol because security analysis of the other two protocols is similar. We demonstrate that our EC-RAC 1 protocol could provide security properties and withstand various attacks.

Authentication: According to the description of our EC-RAC 1 protocol, it is impossible to generate $T_2 = (r_{t1}x(T_1) + r_{s1}x_1r_{t1})Y$ without the secret key x_1 because the adversary faces the ECDL problem. Thus, the server is able to authenticate the tag by checking if $U = r_{s1}^{-1}(y^{-1}T_2 - x(T_1)T_1)$ and x_1T_1 are equal in step 4 of our EC-RAC 1 protocol.

Anonymity: The adversary may intercepts messages $\{T_1\}$, $\{r_{s1}\}$ and $\{T_2\}$ transmitted between the tag and the server, where $T_1 = r_{t1}P$ and $T_2 = (r_{t1}x(T_1) + r_{s1}x_1r_{t1})Y$. Due to the hardness of the ECDL problem, the adversary cannot get any information about x_1 from T_2 because he does not know the server's secret key y. Thus, our EC-RAC 1 could provide anonymity.

Man-in-the-middle attack: Upon receiving the message $\{T_1\}$ generated by the tag, the adversary generates a random number r_a, computes $T_1' = r_a T_1$ and sends message $\{T_1'\}$ to the server, where $T_1 = r_{t1}P$. Upon receiving the message $\{T_1'\}$, the server generates a random number r_{s1}, and sends the message $\{r_{s1}\}$ to the adversary. Upon receiving the message $\{r_{s1}\}$, the adversary sends it to the tag directly. Upon receiving the message $\{r_{s1}\}$, the tag computes $T_2 = (r_{t1}x(T_1) + r_{s1}x_1 r_{t1})Y$ and sends the message $\{T_2\}$ to the server. Upon intercepting the message $\{T_2\}$, the adversary generates a random number r_a, computes $T_2' = r_a T_2$ and sends message $\{T_2'\}$ to the server. Upon receiving the message $\{T_2'\}$, the server computes $U = r_{s1}^{-1}(y^{-1}T_2' - T_1')$. The server checks whether U and $x_1 T_1'$ are equal. It is easy to check that U and $x_1 T_1'$ are not equal. Then, the server could find the attack. Thus, our EC-RAC 1 protocol could withstand the man-in-the-middle attack.

Impersonation attack: The adversary generates a random number r_{t1}, computes $T_1 = r_{t1}P$ and sends the message $\{T_1\}$ the server. Upon receiving $\{T_1\}$, the server generates a random number r_{s1}, and sends the message $\{r_{s1}\}$ to the adversary. However, the adversary cannot generate $T_2 = (r_{t1}x(T_1) + r_{s1}x_1 r_{t1})Y$ because he does not the secret key x_1. The server could find the attack by checking whether $U = r_{s1}^{-1}(y^{-1}T_2 - x(T_1)T_1)$ and $x_1 T_1$ are equal. Thus, our EC-RAC 1 protocol could withstand the impersonation attack.

Replay attack: Suppose the adversary intercepts messages $\{T_1\}$ and $\{T_2\}$ sent by the tag, where $T_1 = r_{t1}P$ and $T_2 = (r_{t1}x(T_1) + r_{s1}x_1 r_{t1})Y$. The adversary replays $\{T_1\}$ to the server. Upon receiving $\{T_1\}$, the server generates a random number r_{s1}, and sends the message $\{r_{s1}\}$ to the adversary. Then, the adversary

replays $\{T_2\}$ to the server. However, the server could find the attack by checking whether $U = r_{s1}^{-1}(y^{-1}T_2 - x(T_1)T_1)$ and x_1T_1 are equal because the server generates a new random number r_{s1} for each session. Thus, our EC-RAC 1 protocol could withstand the replay attack.

Tracking attack: The adversary may intercepts messages $\{T_1\}$, $\{r_{s1}\}$ and $\{T_2\}$ transmitted between the tag and the server, where $T_1 = r_{t1}P$ and $T_2 = (r_{t1}x(T_1) + r_{s1}x_1r_{t1})Y$. However, he cannot get information about tag's identity from those messages because he does not the server's secret key y. Thus, our EC-RAC protocol could withstand the tracking attack.

6. Performance analysis

In this section, we give performance analysis of our three EC-RAC protocols. We also compare the performance of our protocol with that of Lee et al.'s three EC-RAC protocols [21] and Lv et al.'s three EC-RAC protocols [23]. Some notations used in our analysis are defined as follows.

- T_{ma} : the running time of a modular addition operation;
- T_{mm}: the running time of a modular multiplication operation;
- T_{inv} : the running time of a modular inversion operation;
- T_{eca} : the running time an elliptic curve point addition operation;
- T_{ecm}: the running time an elliptic curve point multiplication operation;

Table 1. Computation cost comparison

	The server	The tag
Lee et al.'s EC-RAC 1	$2T_{inv}+1T_{eca}+3T_{ecm}$	$1T_{ma}+1T_{mm}+2T_{ecm}$
Lv et al.'s EC-RAC 1	$2T_{inv}+1T_{eca}+3T_{ecm}$	$1T_{ma}+2T_{mm}+2T_{ecm}$
Our EC-RAC 1	$2T_{inv}+1T_{eca}+4T_{ecm}$	$1T_{ma}+3T_{mm}+2T_{ecm}$
Lee et al.'s EC-RAC 2	$2T_{inv}+2T_{eca}+4T_{ecm}$	$2T_{ma}+3T_{mm}+3T_{ecm}$

Lv et al.'s EC-RAC 2	$2T_{inv}+2T_{eca}+6T_{ecm}$	$2T_{ma}+5T_{mm}+3T_{ecm}$
Our EC-RAC 2	$2T_{inv}+2T_{eca}+7T_{ecm}$	$2T_{ma}+7T_{mm}+3T_{ecm}$
Lee et al.'s EC-RAC 3	$2T_{inv}+2T_{eca}+4T_{ecm}$	$2T_{ma}+3T_{mm}+4T_{ecm}$
Lv et al.'s EC-RAC 3	$2T_{inv}+2T_{eca}+7T_{ecm}$	$2T_{ma}+5T_{mm}+4T_{ecm}$
Our EC-RAC 3	$2T_{inv}+2T_{eca}+8T_{ecm}$	$2T_{ma}+7T_{mm}+4T_{ecm}$

The computational cost comparison of our three EC-RAC protocols, Lee et al.'s three EC-RAC protocols [21] and Lv et al.'s three EC-RAC protocols [23] is demonstrated in Table 1. According to Table 1, the Lee et al.'s EC-RAC 1/2/3 protocol and Lv et al.'s 1/2/3 protocol has better performance than our EC-RAC 1/2/3 protocol. However, Lee et al.'s three EC-RAC protocols [21] suffer from the tracking attack and Lv et al.'s EC-RAC protocols [23] suffer from the man-in-the-middle attack. As a cryptographic protocol, the security is the first important factor in the design of RFID authentication protocol. Our three EC-RAC protocols sacrifice performance slightly to solve the security problems in Lee et al.'s protocols and Lv et al.'s protocol. Therefore, our EC-RAC protocols are more suitable for RFID systems.

7. Conclusions

With the widespread use of the RFID system in our daily life, the design secure RFID authentication protocols attract extensive attention. Recently, ECC-based RFID authentication protocols were studied widely because they could provide better security. Based on Lee et al.'s work, Lv et al. proposed three EC-RAC protocols for authentication in RFID systems. We first demonstrate that Lv et al.'s protocol suffer from the man-in-the-middle attacks. Subsequently, we proposed three security enhanced EC-RAC protocols to solve security problems in Lv et al.'s protocol. Analysis shows that our protocols are more suitable for RFID systems.

Acknowledges

The authors thank the editor and anonymous reviewers for their valuable comments. This study was supported by the National Science foundation of China (Nos. 61402339, 61572370).

References

1. B. Guo, D. Zhang, Z. Yu, Y. Liang, Z. Wang, and X. Zhou, From the Internet of Things to Embedded Intelligence, *World Wide Web Journal*, vol. 16, no. 4, pp. 399-420, 2013.
2. M. Feki, F. Kawsar, M Boussard, and L Trappeniers, The Internet of Things: The Next Technological Revolution, *Computer*, vol. 46, no. 2, pp. 24-25, 2013.
3. R. Das, Rfid market projections 2008–2018, IDTechEx, 2008.
4. Y. Tian, G. Chen, J. Li, A New Ultralightweight RFID Authentication Protocol with Permutation, *IEEE Communication Letters*, vol. 16, no. 5, pp. 702-705, 2012.
5. H. Lee, T. Yi, J. Hyun, Secure and Lightweight Authentication Protocol for Mobile RFID Privacy, *Applied Mathematics & Information Sciences*, vol. 7, no. 1, pp. 421-426, 2013.
6. Y. Lee, Y. Park, A New Privacy-preserving Path Authentication Scheme using RFID for Supply Chain Management, *Advances in Electrical and Computer Engineering*, vol. 13, no. 1, pp. 23-26, 2013.
7. Z. Wu, L. Chen, J. Wu, A Reliable RFID Mutual Authentication Scheme for Healthcare Environments, *Journal of Medical Systems*, vol. 37, no. 2, Article ID: 9917, 2013.
8. C. Yen, M. Lo, N. Lo, Authentication with low-cost RFID tags in mobile networks, *Security and Communication Networks*, vol. 6. no. 8, pp. 1021-1027, 2013.
9. G. Deng, H. Li, Y, Zhang, Tree-LSHB plus : An LPN-Based Lightweight Mutual Authentication RFID Protocol, *Wireless Personal Communication*, vol. 72, no. 1, pp. 159-174, 2013.
10. G. Avoine, M. Bingol, X. Carpent, Privacy-Friendly Authentication in RFID Systems: On Sublinear Protocols Based on Symmetric-Key Cryptography, *IEEE Transactions on Mobile Computing*, vol. 12, no. 10, pp. 2037-2049, 2013.
11. S. Kaul, A. K. Awasthi, RFID Authentication Protocol to Enhance Patient Medication Safety, *Journal of Medical Systems*, vol. 37, no. 6, Article ID: 9979, 2013.
12. M. Dehkordi, Y. Farzaneh, Improvement of the Hash-Based RFID Mutual Authentication Protocol, *Wireless Personal Communication*, vol. 75, no. 1, pp. 219-232, 2014.
13. L. Gao, M. Ma, Y. Shu, An ultralightweight RFID authentication protocol with CRC and permutation, *Journal of Network and Computer Applications*, vol. 41, no. 1, pp. 37-46, 2014.

14. Y. Lee, K. Sakiyama, L. Batina, I. Verbauwhede, Elliptic curve-based security processor for RFID, *IEEE Transactions on Computers*, vol. 57, no. 11, pp. 1514-1527, 2008.
15. Y. Liao, C. Hsiao, A secure ECC-based RFID authentication scheme integrated with ID-verifier transfer protocol, *Ad Hoc Networks*, vol. 18, no. 1, pp. 133-146, 2014.
16. Z. Zhang, Q. Qi, An Efficient RFID Authentication Protocol to Enhance Patient Medication Safety Using Elliptic Curve Cryptography, *Journal of Medical Systems*, vol. 38, no. 5, Article ID: 47, 2014.
17. Z. Zhao, A Secure RFID Authentication Protocol for Healthcare Environments Using Elliptic Curve Cryptosystem, *Journal of Medical Systems*, vol. 38, no. 5, Article ID: 46, 2014.
18. Y. Lee, L. Batina, I. Verbauwhede, EC-RAC (ECDLP based randomized access control): provably secure RFID authentication protocol, In: *IEEE International Conference on RFID 2008*, pp. 97-104, 2008.
19. J. Bringer, H. Chabanne, T. Icart, Cryptanalysis of EC-RAC, a RFID identification protocol. In: *7th International Conference on Cryptology And Network Security-CANS'08*, pp. 149-161, 2008.
20. T. Deursen, S. Radomirovic, Attacks on RFID protocols (version 1.1), *Technical Report*, University of Luxembourg, 2009.
21. Y. Lee, L. Batina, I. Verbauwhede, Untraceable RFID authentication protocols: revision of EC-RAC, In: *IEEE International Conference on RFID 2009*, pp. 178–185, 2009.
22. T. Deursen, S. Radomirovic, Untraceable RFID protocols are not trivially composable: attacks on the revision of EC-RAC. *Technical Report*, University of Luxembourg, 2009.
23. C. Lv, H. Li, J. Ma, Y. Zhang, Vulnerability analysis of elliptic curve cryptography-based RFID authentication protocols, *Transactions on Emerging Telecommunications Technologies*, vol. 23, no. 7, pp. 618–624.
24. S. Vaudenay, On privacy models for RFID, In: *Advances in Cryptology - Asiacrypt 2007*, pp. 68–87, 2007.
25. A. Juels, S. Weis, Defining strong privacy for RFID, *ACM Transactions on Information and System Security*, vol. 13, no. 1, pp. 1–23, 2009.
26. G. Avoine, Adversarial model for radio frequency identification, *Cryptology ePrint Archive*, Report 2005/049, 2005.
27. C. Ng, W. Susilo, Y. Mu, R. Safavi-Naini, RFID privacy models revisited. In: *Proceedings of the 13th European Symposium on Research in Computer Security*, pp. 251–266, 2008.