EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

An elliptic curve cryptosystem-based secure RFID mutual authentication for Internet of things in healthcare environment

Davood Noori[1], Hassan Shakeri[1,2*] and Masood Niazi Torshiz[1,2]

*Correspondence: shakeri@mshdiau.ac.ir

[1] Present Address: Department of Computer Engineering, Sabzevar Branch, Islamic Azad University, Sabzevar, Iran
[2] Department of Computer Engineering, Mashhad Branch, Islamic Azad University, Mashhad, Iran

Abstract

Given the ever-increasing advances in science and technology in recent years, the security and authentication issues using elliptic curve cryptography (ECC) have gained a lot of attentions especially for smart cards in a variety of networks, such as smart homes and medical-based care systems being based on the Internet of things (IoT). In systems such as health care, patient information is always crucial and no one should have access to this information. Recently, much research has been conducted in the area of security and authentication for medical care systems being based on the IoT using RFID technology. One of these schemes is the Alamr et al.'s protocol, which bears high computational cost for authentication between card and reader. In this paper, we have proposed a scheme based on ECC in order to establish a mutual authentication within RFID technology in the IoT. In the proposed scheme, computational cost, communication cost and elliptic curve point multiplication running time and data storage cost are investigated. Moreover, security requirements and various attacks have also been considered in the proposed protocols. AVISPA software has also been used for security analysis. The analyses represent that the proposed scheme has lower computational costs, lower communication costs and less elliptic curve point multiplication time compared with those of similar protocols. It has also resolved the security shortcomings of the RFID authentication protocol.

Keywords: Elliptic curve cryptography (ECC), Internet of things (IoT), RFID, Healthcare, Mutual authentication

1 Introduction

The debate on the Internet technology of objects is steadily increasing, and numerous research works have been done on this technology. This technology introduces an intelligent management and has a variety of models, including those models having many uses such as radio waves and bar codes [1]. The IoT these days has created a situation in which the lifeless things around us interact with and know each other [2]. These physical objects can be interconnected in a variety of ways, and some of them are: Wireless Fidelity, Radio Frequency Identification, and Quick Response Code [1]. The IoT has many uses in our society and today's life, including smart homes, smart cities, industrial Internet, connected machines, smart networks, intelligent retail, smart health, etc. The devices used in IOT communicate by radio communications, and security in such a network is low. However, the data transmitted in theme are of high importance; therefore, it is essential to provide a solid security scheme for these networks.

The RFID, discussed in research work, uses the wireless communication channels between cards and card readers; depending on the type of communication, it is determined that the transmitting space of spatial information is completely open and vulnerable and numerous attacks can threaten this connection; some of these probable attacks can be: eavesdropping, tracking and network scanning [3]. Therefore, there are many ways to secure this technology. An advantage of RFID technology is that if there is a direct distance, the objects can be identified and interconnected without contacting each other; this can be done using radio waves. Generally, a complete RFID system consists of three parts: card, card reader and a backup server or database. The cards can be divided into two active and passive categories according to their power supply; in addition, cards can use low-frequency waves, high-frequency waves or super-high-frequency waves. [3].

Several articles have used various cryptographies to provide security for RFID technology, some of them have used ECC, which is of asymmetric algorithms [4–8], and some of them are based on symmetric algorithms [9, 10]. ECC is a public key encryption approach based on the algebraic structure of elliptic curves in finite fields. ECCs also require smaller keys to provide relative security; this is the reason for its superiority [11, 12]. For current cryptographic purposes, an elliptic curve has been placed on a finite field (instead of actual numbers), where the points of interest have been specified along an infinite point. The use of an elliptic curve for cryptography was proposed by Neal Koblitz and Victor S. Miller in 1985. The elliptic curve started to be widely used in cryptography between 2004 and 2005 [13, 14]. The first RFID authentication protocol, being based on the elliptic curve algorithm, was presented by Tuyls and Banita [15]. Banita et al. presented a similar scheme of elliptic curve in 2007 using the Okamoto authentication scheme, but their protocols had problems that violated privacy [1]. Recently, papers such as Alamr et al. [3] and Liao [16] have used ECC for RFID authentication; however, they have had many computational costs. Another study proposed by Yang [17] addresses the problems of the Kaur [18] scheme which has a high computational cost. Yang [17] has made changes to the Kaur scheme and presented a new scheme reducing the high cost of computation.

A recent study done by Sowjanya et al. [19] has reviewed protocol proposed by Li et al. [20]. They have reviewed Li's "ECC" and then questioned security problems such as man in the middle and lack of mutual authentication. Finally, they have proposed a new authentication protocol using ECC for monitoring of the healthcare system, having a lower computational cost. It has three phases: initialization, registration and authentication. Although it is a good protocol, it has a high computational cost. Dang et al. [21] who has developed Wang et al. protocol [22] have utilized ECC and cloud servers. They claimed that their scheme is energy efficient, but it has high computational cost. Li et al. [23] have also proposed another protocol for mutual authentication in distant healthcare system using cloud servers, but their scheme has high communicative cost. It also suffers from some security problems like forging attack, lack of message authentication, and session key threatening. Kumara et al. [24] also proposed another protocol. They have reviewed and discussed Li et al. [23]

security problem scheme. They have decreased the communication cost and also resolved the security problems.

Recently, another protocol has been proposed by Noori et al. [25] which has a lower computational cost rather than other schemes. Furthermore, unlike most of the previous works, this protocol has 3 phases including: initialization, authentication and scalability phase. Some of key advantages of this protocol are that: This protocol is faster than other ones, and various attacks are considered in this protocol. Chinnasamy et al. [26] also proposed another protocol. They combine elliptic curve and Blowfish cryptography. In the combined algorithm of these authors, it is claimed that it has high confidentiality and security for patient data. In another protocol proposed by Dhanda et al. [27], the method of ECC has been compared with other Black Cypher methods, in which ECC and AES have been the most suitable lightweight cryptographic methods.

The structure of the paper is organized as follows: In subsection 1.2, first**,** we review A.A. Alamr et al. scheme. Next, in Sect. 2 we discussed about the method and the proposed scheme, including the initialization phase and the authentication phase; then, the analysis of the proposed solution as well as security and its efficiency have been addressed in Sect. 3, and finally, we presented the conclusions in Sect. 4.

1.1 Review of Alamr's scheme

The Alamr et al. [3]. scheme is made up of two phases of initialization and authentication. The signs are shown in Table 1 to have a better view.

1.1.1 Initialization phase

In this phase, the server generates system parameters. To do this, the server selects a random number $P_{r_R} \in F_P$ as the private key of the card reader, and then, using Eq. (1), it generates a public key for the card reader:

$$P_{u_R} = P_{r_R}P \tag{1}$$

In addition, the server selects a random number $P_{r_T} \in F_P$ as the private key of card, and then, using Eq. (2), it generates a public key for the card as well:

$$P_{u_T} = P_{r_T}P \tag{2}$$

Table 1 Definition of notations used in the Alamr scheme [3]

Notation	Definition
GF(p)	Galois field
N	Elliptic curve order
P	Elliptic curve base point
a,b	Cofactors of elliptic curve equation "part of the ECC common parameters"
P_{r_R}	Private key card reader
P_{u_R}	Public key card reader
P_{r_T}	Private key card
P_{u_T}	Public key card

Then, the card and the card reader store their pair of keys in their memories, as well as the system parameters and parameters required in Table 2.

1.1.2 Authentication phase

This phase includes the following steps:

Step 1 The card reader generates a random number that is given in Eq. (3) and then calculates a point on an elliptic curve such as Eq. (4), and then, the card reader sends R_1 to the card.

$$r_1 \in F_P \tag{3}$$

$$R_1 = r_1.P \tag{4}$$

Step 2 Once R_1 is received by card from the card reader, it generates a random number mentioned in Eq. (5) and then calculates a point on the elliptic curve such as Eq. (6). In the next step, the card creates 2 secret key using Eq. (7); then, the secret keys are encrypted using Eq. (8), and in the final step, the card sends C_1, T_1 to the card reader.

$$t_1 \in F_P \tag{5}$$

$$T_1 = t_1.P \tag{6}$$

$$SK1_T = P_{r_T}.R_1, SK2_T = t_1.R_1 \tag{7}$$

$$C_1 = SK1_T + SK2_T \tag{8}$$

Step 3 Once T_1 and C_1 are received, the card reader generates two temporary secret keys to reconstruct the encryption keys shown in (9) and then calculates Eq. (10) and compares it with Eq. (8); if they are the same, then the card will be authenticated (11). Next, the card reader calculates Eq. (12); in addition, it produces a new random number (13) and an elliptic curve point (14) for being used in the key agreement. After that, the card reader sends C_2, R_2 to the card.

$$SK1_R = r_1.P_{u_T}, SK2_R = r_1.T_1 \tag{9}$$

$$X = SK1_R + SK2_R \tag{10}$$

$$X = C_1 \tag{11}$$

$$C_2 = T_1.P_{r_R} \tag{12}$$

Table 2 System parameters Alamr scheme [3]

System parameters	P_{U_R}, P, n
Card reader storage	$P_{U_R}, P_{r_R}, P_{U_T}, P, n$
Card storage	$P_{U_T}, P_{r_T}, P_{U_R}, P, n$

$$r_2 \in F_P \tag{13}$$

$$R_2 = r_2 P \tag{14}$$

Step 4 The card calculates Eq. (15) and compares it with (12); if the result is the same, then the card reader is also authenticated (16).

$$Y = t_1.P_{u_R} \tag{15}$$

$$C_2 = Y \tag{16}$$

Step 5 Then, the agreed keys are formed between the two sides, the card's key is equal to Eq. (17) and the card reader's is equal to (18).

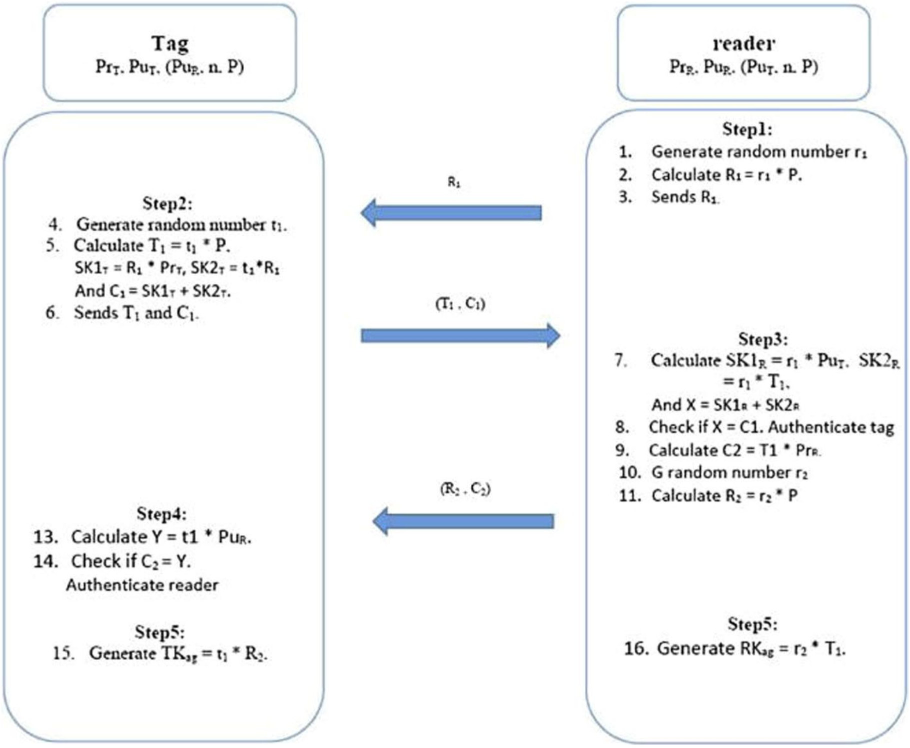
$$TK_{ag} = t_1.R_2 \tag{17}$$

$$RK_{ag} = r_2.T_1 \tag{18}$$

1.1.3 The costs of A.A. Alamr's scheme

The costs of this scheme as indicated by Alamr itself have 3 random numbers, 9 multiplications and 2 additions of the elliptic curve points, without hashing process.

The authentication phase of Alamr scheme [3] is illustrated in Fig. 1.


Tag
Pr_T. Pu_T. (Pu_R. n. P)
reader
Pr_R. Pu_R. (Pu_T. n. P)
R_1
(T_1 , C_1)
(R_2 , C_2)

Step1:

1. Generate random number r_1
2. Calculate $R_1 = r_1 * P$.
3. Sends R_1.

Step2:

4. Generate random number t_1.
5. Calculate $T_1 = t_1 * P$.
   $SK1_T = R_1 * Pr_T$, $SK2_T = t_1 * R_1$
   And $C_1 = SK1_T + SK2_T$.
6. Sends T_1 and C_1.

Step3:

7. Calculate $SK1_R = r_1 * Pu_T$. $SK2_R = r_1 * T_1$.
   And $X = SK1_R + SK2_R$
8. Check if X = C1. Authenticate tag
9. Calculate C2 = T1 * Pr_R.
10. G random number r_2
11. Calculate $R_2 = r_2 * P$

Step4:

13. Calculate Y = t1 * Pu_R.
14. Check if $C_2 = Y$.
    Authenticate reader

Step5:

15. Generate $TK_{ag} = t_1 * R_2$.

Step5:

16. Generate $RK_{ag} = r_2 * T_1$.

Fig. 1 The authentication protocol of Alamr scheme [3]

Table 3 Definition of notations used in the proposed scheme

Notation	Definition
GF(p)	Galois field
N	Elliptic curve order
P	Elliptic curve base point
a,b	Cofactors of elliptic curve equation "part of the ECC common parameters"
P_{r_R}	Private key card reader
P_{u_R}	Public key card reader
P_{r_T}	Private key card
P_{u_T}	Public key card
H	Hash function

Table 4 System parameters

System parameters	P_{u_R}, P, n, H
Card reader storage	$P_{u_R}, P_{r_R}, P_{u_T}, P, n, H$
Card storage	$P_{u_T}, P_{r_T}, P_{u_R}, P, n, H$

2 Methods

Lately, a scheme has been proposed by Alamr et al. [3], which has used an ECC algorithm for RFID authentication, being described as a good scheme. However, after having studied and reviewed this reference carefully, we found that the proposed scheme had higher computational costs, higher communication cost, and higher elliptic curve point multiplication running time while operating.

Thus, we propose a scheme based on the present research that has two phases: the Initialization phase and the authentication phase. Our scheme, providing the necessary security, has several benefits: It has reduced the computational and communication cost and lowered elliptic curve point multiplication running time. The Alamr scheme is discussed in detail in Sect. 1.2.

The signs are shown in Table 3 to have a better view.

2.1 Initialization phase

In this phase, the server generates system parameters. To do this, the server selects a random number $P_{r_R} \in F_P$ as the private key of the card reader. Then, a public key is generated for the card reader using Eq. (19) as follows:

$$P_{u_R} = P_{r_R}P \tag{19}$$

The server also selects a random number $P_{r_T} \in F_P$ as the private key of the card. Then, a public key is generated for the card using Eq. (20):

$$P_{u_T} = P_{r_T}P \tag{20}$$

Then, the card and card reader store their pair of keys with the system parameters and parameters required in Table 4 in their memory.

2.2 Authentication phase

This phase includes the following steps:

Step 1 The card reader generates a random number produced by Eq. (21) and next calculates a point on the elliptic curve, such as Eq. (22), and then, R is sent to the card by the card reader.

$$r \in F_P \tag{21}$$

$$R = r.P \tag{22}$$

Step 2 After the card receives R from the card reader, it generates a random number being mentioned in Eq. (23) and then calculates a point on the elliptic curve such as Eq. (24). In the next step, the card creates a secret key through Eq. (25), and finally, the card calculates Eq. (26) for the encryption of the secret key and then sends C_1, T to the card reader.

$$t \in F_P \tag{23}$$

$$T = t.P \tag{24}$$

$$SK1_T = R.P_{r_T} \tag{25}$$

$$C_1 = H(SK1_T, T, R) \tag{26}$$

Step 3 After receiving T and C_1, the card reader generates a secret key for reconstructing the encrypted key shown in Eq. (27),and then, it calculates Eq. (28) and compares it with Eq. (26). If they were identical, then the card is authenticated (29). Next, the card reader creates a secret key given in Eq. (30) and then calculates Eq. (31). After that, the card reader sends C_2 to the card.

$$SK2_T = r.P_{u_T} \tag{27}$$

$$X = H(SK2_T, T, R) \tag{28}$$

$$X = C_1 \tag{29}$$

$$SK1_R = T.P_{r_R} \tag{30}$$

$$C_2 = H(SK1_R, X) \tag{31}$$

Step 4 The card calculates the secret key using Eq. (32) and uses it to calculate Eq. (33) and then compares it with Eq. (31); if they are the same, the card reader is authenticated as well (34).

$$SK2_R = t.P_{u_R} \tag{32}$$

$$Y = H(SK2_R, C_1) \tag{33}$$

$$C_2 = Y \tag{34}$$

Step 5 Then, agreed keys are formed between the two sides, the side of the card is equal to Eq. (35) and the side of the card reader is equal to Eq. (36).

$$TK_{ag} = t.R \tag{35}$$

$$RK_{ag} = r.T \tag{36}$$

The authentication phase of proposed scheme is illustrated in Fig. 2.

2.3 The costs of proposed scheme

As it is known, the proposed scheme has two random numbers, 6 multiplications as well as 4 hashing processes.

3 Security and performance results and discussion

3.1 Security

In this section, we compare the proposed scheme with recent articles based on some of the important security requirements, such as mutual authentication, confidentiality,

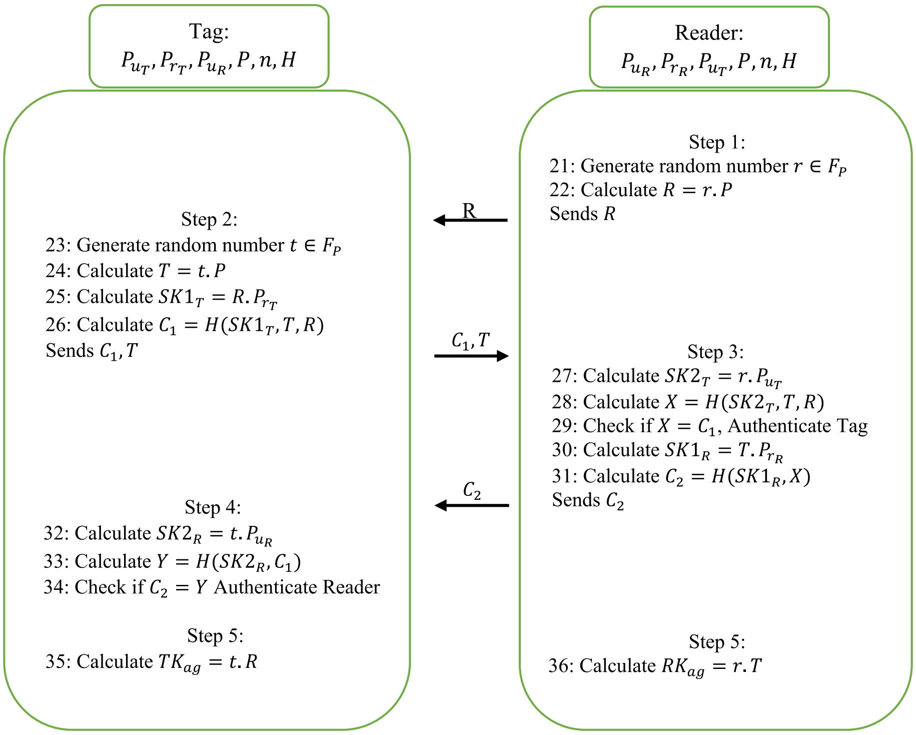

Tag:
$P_{u_T}, P_{r_T}, P_{u_R}, P, n, H$
Reader:
$P_{u_R}, P_{r_R}, P_{u_T}, P, n, H$
Step 1:
21: Generate random number $r \in F_P$
22: Calculate $R = r.P$
Sends R
R
Step 2:
23: Generate random number $t \in F_P$
24: Calculate $T = t.P$
25: Calculate $SK1_T = R.P_{r_T}$
26: Calculate $C_1 = H(SK1_T, T, R)$
Sends C_1, T
C_1, T
Step 3:
27: Calculate $SK2_T = r.P_{u_T}$
28: Calculate $X = H(SK2_T, T, R)$
29: Check if $X = C_1$, Authenticate Tag
30: Calculate $SK1_R = T.P_{r_R}$
31: Calculate $C_2 = H(SK1_R, X)$
Sends C_2
C_2
Step 4:
32: Calculate $SK2_R = t.P_{u_R}$
33: Calculate $Y = H(SK2_R, C_1)$
34: Check if $C_2 = Y$ Authenticate Reader
Step 5:
35: Calculate $TK_{ag} = t.R$
Step 5:
36: Calculate $RK_{ag} = r.T$

Fig. 2 The proposed authentication protocol

anonymity, availability, data integrity, privacy and forward security; moreover, the resistance of the proposed scheme to some major attacks has been scrutinized, including man-in-the-middle attack, replay attack, DoS attack and forging attack. Furthermore, in order to confirm the security of the proposed scheme against active and inactive attacks, we will use AVISPA software and HLPSL descriptor language.

In these networks, forging attacks are hard to detect, as the ID of the licensed card is forged in this attack. Now, let us assume the attacker is trying to get the card's secret key; to do this the attacker should solve the discrete logarithm for the elliptic curve. Given the resistance of the discrete logarithm for the elliptic curve, as outlined in Johnson [28] and Barker [29], the proposed scheme will be reliable against this attack.

3.1.1 Man-in-the-middle attack

In this attack, the attacker tries to destroy the communication channel between the card and the card reader. In this case, the attacker cannot extract any useful information that triggers the attack. If we assume that the attacker gets C_1 while exchanging, based on ECDLP, the attacker cannot get the private key to use it and communicate with the card reader and cannot calculate the correct X value if he/she uses an invalid private key for the card reader.

3.1.2 Replay attack

In this attack, the attacker tries to use a repeated message in the network and perform the authentication process. Since the random number is generated for each connection, the attacker cannot use repeated messages. As an example, imagine an attacker has received all the messages exchanged between the card and the card reader in one session. Now, the attacker wants to prove itself as permitted card in the new session by T, C_1 parameters from the previous session. The card reader then will be able to identify the attacker by C_1.

$$X \neq C_1,$$

$$H(SK2_T, T, R) \neq H(SK1_T, T, R),$$

$$H\left(\overbrace{r_{\text{new}}.P.P_{u_T}}^{\checkmark}, \overbrace{t_{\text{old}}.P}^{\checkmark}, \overbrace{r_{\text{new}}.P}^{\checkmark}\right) \neq H\left(\overbrace{r_{\text{old}}.P.P_{r_T}}^{\times}, \overbrace{t_{\text{old}}.P}^{\checkmark}, \overbrace{r_{\text{old}}.P}^{\times}\right)$$

Figure 3 shows illegal card identification. Also, if the attacker wants to prove itself by C_2, R from the previous session as permitted card reader in new session, the card will be able to identify it by C_2 parameter.

$$Y \neq C_2$$

$$H(SK2_R, C_1) \neq H(SK1_R, X)$$

$$H(t.P_{u_R}, H(SK1_T, T, R)) \neq H(T.P_{r_R}, H(SK2_T, T, R))$$

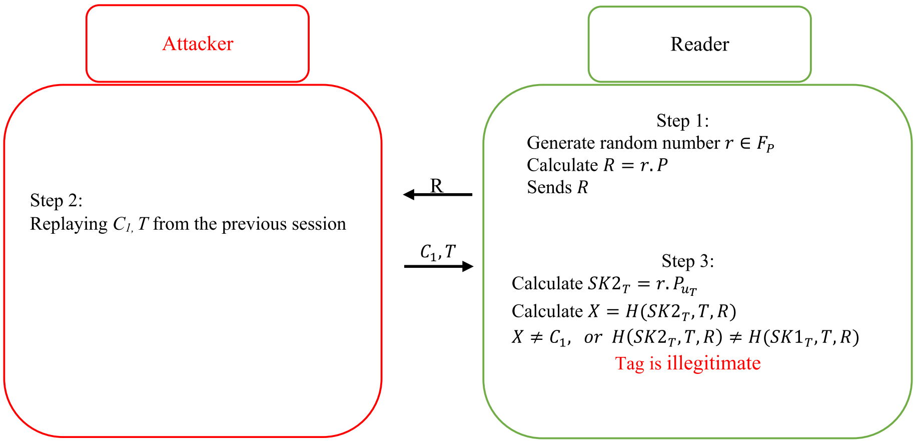

Attacker
Reader
Step 1:
Generate random number $r \in F_P$
Calculate $R = r.P$
Sends R
R
Step 2:
Replaying $C_{1,}\ T$ from the previous session
C_1, T
Step 3:
Calculate $SK2_T = r.P_{u_T}$
Calculate $X = H(SK2_T, T, R)$
$X \neq C_1,\ or\ H(SK2_T, T, R) \neq H(SK1_T, T, R)$
Tag is illegitimate

Fig. 3 Illegal card identification

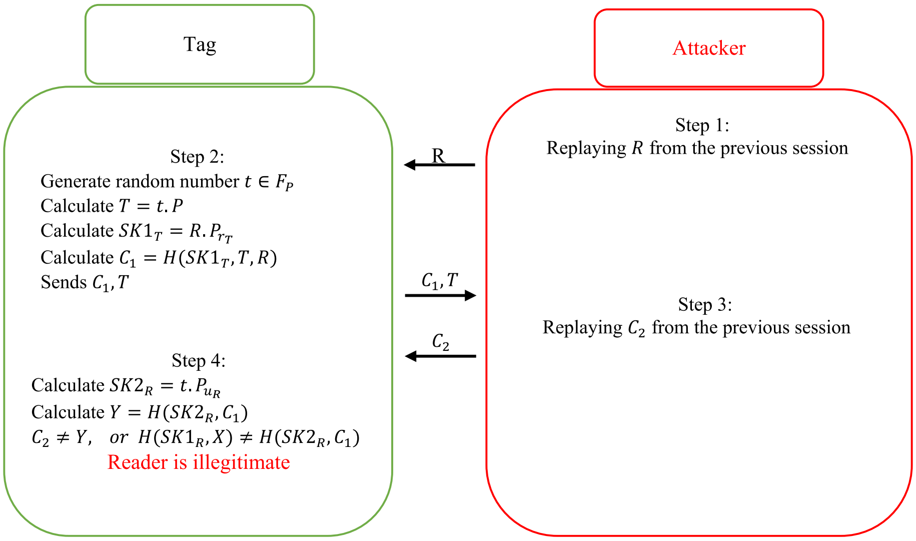

Tag
Attacker
Step 1:
Replaying R from the previous session
R
Step 2:
Generate random number $t \in F_P$
Calculate $T = t.P$
Calculate $SK1_T = R.P_{r_T}$
Calculate $C_1 = H(SK1_T, T, R)$
Sends C_1, T
C_1, T
Step 3:
Replaying C_2 from the previous session
C_2
Step 4:
Calculate $SK2_R = t.P_{u_R}$
Calculate $Y = H(SK2_R, C_1)$
$C_2 \neq Y,\ \ or\ \ H(SK1_R, X) \neq H(SK2_R, C_1)$
Reader is illegitimate

Fig. 4 Illegal card reader identification

$$H(t.P_{u_R}, H(R.P_{r_T}, T, R)) \neq H(T.P_{r_R}, H(r.P_{u_T}, T, R))$$

$$H\left(\overbrace{t_{\text{new}}.P_{u_R}}^{\checkmark}, H\left(\overbrace{r_{\text{old}}.P.P_{r_T}}^{\checkmark}, \overbrace{t_{\text{new}}.P}^{\checkmark}, \overbrace{r_{\text{old}}.P}^{\checkmark}\right)\right) \neq H\left(\overbrace{t_{\text{old}}.P.P_{r_R}}^{\times}, H\left(\overbrace{r_{\text{old}}.P_{u_T}}^{\checkmark}, \overbrace{t_{\text{old}}.P}^{\times}, \overbrace{r_{\text{old}}.P}^{\checkmark}\right)\right)$$

As a result, the proposed protocol is resistant to replay attack. Figure 4 shows illegal card reader identification.

3.1.3 Mutual authentication

In the proposed protocol, if the value of X is equal to C_1, the card reader will be able to authenticate the card:

$$X = C_1,$$

$$H(SK2_T, T, R) = H(SK1_T, T, R),$$

$$H\left(\overbrace{r.P.P_{u_T}}^{\checkmark}, \overbrace{t.P}^{\checkmark}, \overbrace{r.P}^{\checkmark}\right) = H\left(\overbrace{r.P.P_{r_T}}^{\checkmark}, \overbrace{t.P}^{\checkmark}, \overbrace{r.P}^{\checkmark}\right)$$

In addition, the card can also validate the card reader using Y and C_2 values if these two values are equal:

$$Y = C_2$$

$$H(SK2_R, C_1) = H(SK1_R, X)$$

$$H(t.P_{u_R}, H(SK1_T, T, R)) = H(T.P_{r_R}, H(SK2_T, T, R))$$

$$H(t.P_{u_R}, H(R.P_{r_T}, T, R)) = H(T.P_{r_R}, H(r.P_{u_T}, T, R))$$

$$H\left(\overbrace{t.P_{u_R}}^{\checkmark}, H\left(\overbrace{r.P.P_{r_T}}^{\checkmark}, \overbrace{t.P}^{*}, \overbrace{r.P}^{\checkmark}\right)\right) = H\left(\overbrace{t.P.P_{r_R}}^{\checkmark}, H\left(\overbrace{r.P_{u_T}}^{\checkmark}, \overbrace{t.P}^{\checkmark}, \overbrace{r.P}^{\checkmark}\right)\right)$$

3.1.4 Confidentiality

Based on ECDLP, the attacker cannot retrieve the private key from the messages.

3.1.5 Anonymity

It can be said that anonymity is one of the subtypes of confidentiality. Since a new random number is generated at each step, the attacker cannot guess the private key and ID.

3.1.6 Forward security

If we assume that the attacker has acquired the private and public keys of the card with a physical attack, he cannot predict the previously exchanged messages of the card, because he does not know the random number generated and used.

3.1.7 Location privacy

The variety of messaging between the card and the card reader, as well as the rules in the ECDLP, make it very difficult for the attacker to track the card.

3.1.8 Data integrity

In data integrity, the data exchanged between card reader and card must not be changed by the attacker. In the proposed protocol, public and private keys have already been loaded in both card reader and card and keys including $C_1, C_2, SK1_T, SK2_T, SK1_R, SK2_R$ have been generated by them at the authentication phase. This means that any change done by attacker on encrypted data will mismatch at the receiver side. Therefore, the attack will be detected. Because of this, the proposed protocol is strong enough against this attack and can guarantee the data integrity.

3.1.9 Denial of service attack

As no secret keys in card or even card reader in our proposed protocol is updated concurrently, it is then resistant to DoS attack [25, 30]. Also, imagine an attacker wants to down the authentication between the card and the card reader by DoS attack. As our proposed scheme has mutual authentication, it is then able to resist to DoS attack. To do that, the attacker must in fact change one of these keys: $SK1_T, SK2_T, SK1_R, SK2_R$. However, this is an impossibility because of ECDLP. As a result, DoS attack is not possible on our proposed protocol.

3.1.10 Availability

As mentioned before, no secret keys in our proposed protocol are updated synchronously, and it can then run between the card and the card reader and is also available. Table 5 shows the security comparison between related protocols and the proposed method:

3.2 Performance

To evaluate the performance, we have compared our proposed method with similar recent research papers such as Alamr [3], Liao [16], Sowjanya [19] and Dang [21]. We estimated the computation costs, communication costs, data storage costs and the execution time of the elliptical curve point multiplication of authentication phase for each of the five methods, for both the card reader and card. In Table 6, we present some of the various symbols used in this section.

On the other hand, according to the methods proposed in NikooGhadam et al. and Koblitz et al. [31, 32], the complexity of time for implementation of various operational phases has been calculated using modular exponentiation. The results are specified in Table 7.

Table 5 Security comparison among related protocols

Item	Proposed scheme	Alamr [3]	Liao [16]	Sowjanya [19]	Dang [21]
Confidentiality	YES	YES	YES	YES	YES
Data integrity	YES	No	No	–	–
Availability	YES	No	YES	No	No
Anonymity	YES	YES	No	YES	No
Forwards security	YES	YES	YES	YES	YES
Location privacy	YES	YES	YES	No	No
Avoiding forgery attack	YES	YES	YES	YES	YES
Avoiding replay attack	YES	YES	YES	YES	YES
Mutual authentication	YES	YES	YES	YES	YES
Avoiding Man-in-the-middle attack	YES	YES	No	YES	YES
Avoiding DoS attack	YES	No	YES	YES	No

Table 6 Definition of some notations used in performance evaluation of the proposed scheme

Notation	Definition
T_H	Hash computation time
T_E	Elliptic curve polynomial computation time
T_{PA}	Elliptic curve point addition computation time
T_{PM}	Elliptic curve point multiplication computation time
T_{PR}	Private key computation time
T_{PU}	Public key computation time

The Hash computation cost (T_H) is much less than the private and public keys computation costs (T_{PU} and T_{PR})

Table 7 Unit conversion of various operations in terms of T_{MUL}

Time complexity of an arithmetic unit	Time complexity in terms of modular multiplication
T_{PM}	1200 T_{MUL}
T_{PA}	5 T_{MUL}
T_H	Negligible

Table 8 Performance comparison of computation costs among related protocols in authentication phase

	Time complexity	Time complexity in unit of T_{MUL}
Liao [16]	$10T_{PM} + 4T_{PA}$	$[12000 + 20]T_{MUL}$
Alamr [3]	$9T_{PM} + 2T_{PA}$	$[10800 + 10]T_{MUL}$
Sowjanya [19]	$9T_{PM} + 2T_H$	$[10800]T_{MUL}$
Dang [21]	$8T_{PM} + 9T_H$	$[9600]T_{MUL}$
Proposed scheme	$6T_{PM} + 4T_H$	$[7200]T_{MUL}$

The comparison between five methods is shown in Table 8. Since all of the public and private keys and other main parameters are loaded into the card and card reader in the initialization phase, thus, the computational cost of the private and public keys and other parameters are zero. In the authentication phase of our method, the computational cost is $4\,T_H + 6\,T_{PM}$ for the card and card reader. Therefore, our proposed method has lower computational cost, when compared to other methods.

It is also possible to calculate the execution time of the most complex operations on elliptic curve, that is, the elliptic curve point multiplication in milliseconds. For instance, we assume that all of the related articles use an elliptic curve with an equal key length of 160 bits. The execution time of the elliptical curve point multiplication on 5 MHZ cards equals 0.064 s [33]. The running time of the elliptic curve point multiplication among the related protocols for both the card and the card reader is given in Table 9.

Also, in proposed protocol, the communication cost between tag and card reader at the authentication phase has been calculated by the message length calculation which was sent. In our proposed protocol, messages C_1, R, T,C_2 are exchanged between card and card reader at the authentication phase. As it is known that the elliptic curve has X and Y coordinates. If we imagine the elliptic curve length is

Table 9 The running time of the elliptic curve point multiplication among related protocols in authentication phase

	Running time of the elliptic curve point multiplication		**Total**
	Tag	**Reader**	
Liao [16]	$5T_{PM} = 5 * 64 = 320$	$5T_{PM} = 5 * 64 = 320$	$640(ms)$
Alamr [3]	$4T_{PM} = 4 * 64 = 256$	$5T_{PM} = 5 * 64 = 320$	$576(ms)$
Sowjanya [19]	$3T_{PM} = 3 * 64 = 192$	$6T_{PM} = 6 * 64 = 384$	$576(ms)$
Dang [21]	$5T_{PM} = 5 * 64 = 320$	$3T_{PM} = 3 * 64 = 192$	$512(ms)$
Proposed scheme	$3T_{PM} = 3 * 64 = 192$	$3T_{PM} = 3 * 64 = 192$	$384(ms)$

160 bits, we will then have 320 bits. Consequently, our communication cost will be: $320 + 160 + 160 + 320 = 960$, whereas Alamr's communication cost sum is $320 + 320 + 320 + 320 + 320 = 1600$. Table 10 shows list of variables and their values in bits.

The communication cost among related protocols in authentication phase is shown in Table 11.

We have also calculated information storage cost for card reader and card in our proposed protocol. Cards must store parameters like $P_{u_T}, P_{r_T}, P_{u_R}, P, n, H, a, b, q$ as they do in Alamr's scheme. A parameter named H has been added to it, and the sum is then 2080 bits. Card reader's cost is exactly similar to Alamr's scheme. The only difference is that parameter H has been added to it. Parameter J indicates how many cards there are. Table 12 shows information storage cost among related protocols.

Table 10 List of variables and their values

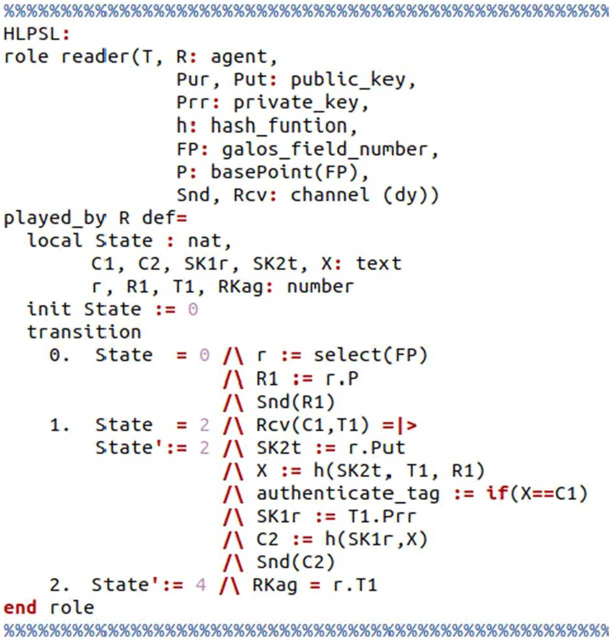
Notation	**values (bits)**
Common parameters:a, b, q, n	160,160,160,160
P : basepointonellipticcurve	320
Private key of the server/reader	160
Public key of the server/reader	320
Private key of the tag	160
Public key of the tag	320
Hash function	160
Symmetric encryption/decryption/session keys	128

Table 11 Comparison of communication cost among related protocols in authentication phase

	Tag (bits)	**Reader (bits)**	**Total communication cost**
Liao [16]	640	640	1280
Alamr [3]	640	960	1600
Sowjanya [19]	640	480	1120
Dang [21]	1248	128	1376
Proposed scheme	480	480	960

Table 12 Comparison of data storage cost among related protocols in authentication phase

	Tag (bits)	Reader (bits)	Total storage cost
Liao [16]	1760	1120 + 480j	2880 + 480j
Alamr [3]	1920	1120 + 320j	3040 + 320j
Sowjanya [19]	2368	1568 + 320j	3936 + 320j
Dang [21]	1600	1760 + 320j	3360 + 320j
Proposed scheme	2080	1280 + 320j	3360 + 320j



```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
HLPSL:
role reader(T, R: agent,
               Pur, Put: public_key,
               Prr: private_key,
               h: hash_funtion,
               FP: galos_field_number,
               P: basePoint(FP),
               Snd, Rcv: channel (dy))
played_by R def=
  local State : nat,
       C1, C2, SK1r, SK2t, X: text
       r, R1, T1, RKag: number
  init State := 0
  transition
    0.   State  = 0 /\ r := select(FP)
                    /\ R1 := r.P
                    /\ Snd(R1)
    1.   State  = 2 /\ Rcv(C1,T1) =|>
         State':= 2 /\ SK2t := r.Put
                    /\ X := h(SK2t, T1, R1)
                    /\ authenticate_tag := if(X==C1)
                    /\ SK1r := T1.Prr
                    /\ C2 := h(SK1r,X)
                    /\ Snd(C2)
    2.  State':= 4 /\ RKag = r.T1
end role
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

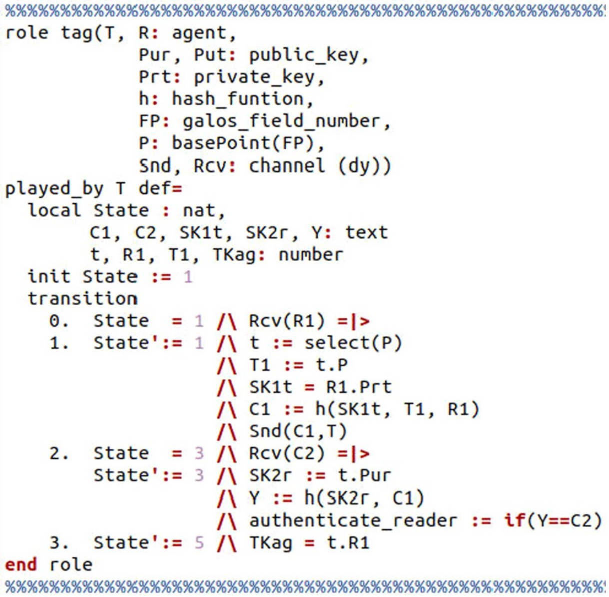
Fig. 5 Authentication steps for the card reader in HLPSL

3.3 Simulation using AVISPA tool

The AVISPA simulation tool is an extensively used security verification software which checks whether the authentication protocol is SAFE or UNSAFE against any active and passive attacks. The high-level language supported by this tool is High-Level Protocol Specification Language (HLPSL) [19].

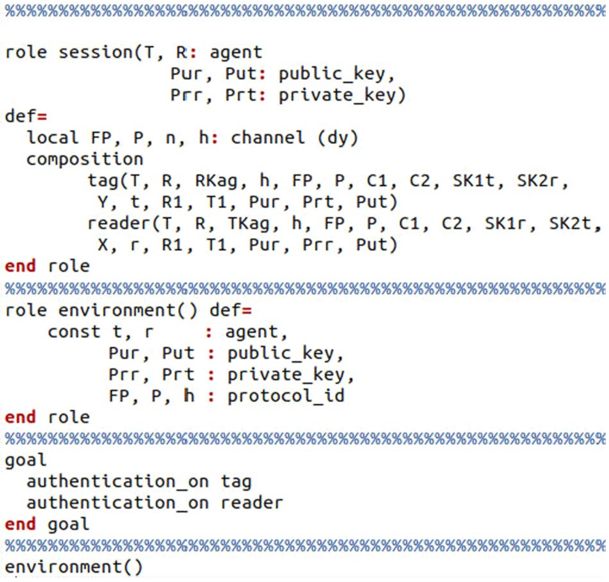
AVISPA software is deployed as a simulator to assess the security of different protocols and determine whether or not different security protocols are secure. For the proposed scheme, the authentication steps for all the main roles, the card and the card reader have been written in HLPSL, and finally, the input codes were entered into the AVISPA software. The output of the proposed scheme is secure confirming the security of the proposed scheme against various attacks. The role of writing for the card reader is shown in Fig. 5.

The role of writing for the card is shown in Fig. 6; in the end, the characteristics of the roles and the working session between them for authentication are given in Fig. 7.



```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
role tag(T, R: agent,
            Pur, Put: public_key,
            Prt: private_key,
            h: hash_funtion,
            FP: galos_field_number,
            P: basePoint(FP),
            Snd, Rcv: channel (dy))
played_by T def=
  local State : nat,
        C1, C2, SK1t, SK2r, Y: text
        t, R1, T1, TKag: number
  init State := 1
  transition
    0.  State  = 1 /\ Rcv(R1) =|>
    1.  State':= 1 /\ t := select(P)
                   /\ T1 := t.P
                   /\ SK1t = R1.Prt
                   /\ C1 := h(SK1t, T1, R1)
                   /\ Snd(C1,T)
    2.  State  = 3 /\ Rcv(C2) =|>
        State':= 3 /\ SK2r := t.Pur
                   /\ Y := h(SK2r, C1)
                   /\ authenticate_reader := if(Y==C2)
    3.  State':= 5 /\ TKag = t.R1
end role
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Fig. 6 Authentication steps for the card in HLPSL

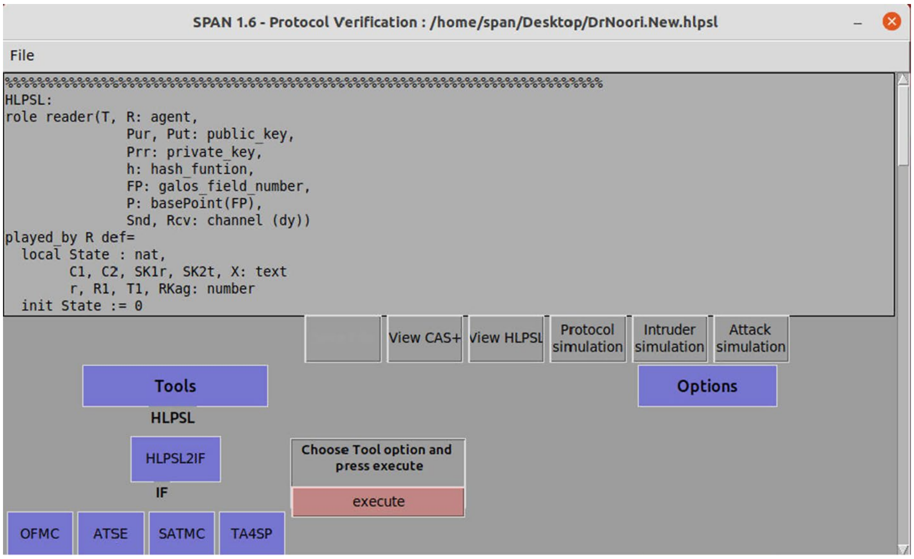


```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

role session(T, R: agent
               Pur, Put: public_key,
               Prr, Prt: private_key)
def=
  local FP, P, n, h: channel (dy)
  composition
        tag(T, R, RKag, h, FP, P, C1, C2, SK1t, SK2r,
         Y, t, R1, T1, Pur, Prt, Put)
        reader(T, R, TKag, h, FP, P, C1, C2, SK1r, SK2t,
         X, r, R1, T1, Pur, Prr, Put)
end role
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
role environment() def=
    const t, r     : agent,
          Pur, Put : public_key,
          Prr, Prt : private_key,
          FP, P, h : protocol_id
end role
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
goal
  authentication_on tag
  authentication_on reader
end goal
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
environment()
```

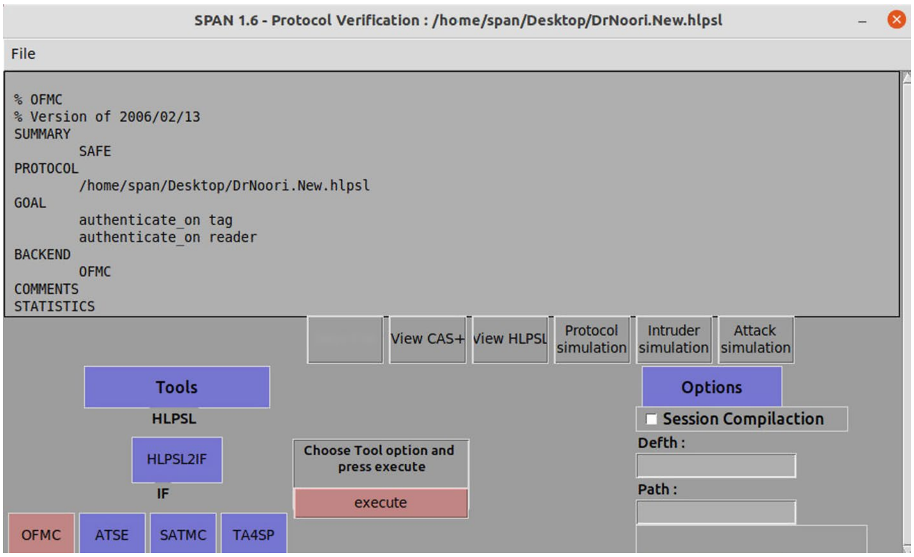
Fig. 7 The characteristics of the roles and the working session between card and card reader for authentication

Then, for each role, the written codes were entered into AVISPA software and a safe output was obtained by the software. Figure 8 shows the entered codes into the AVISPA software, and Figs. 9 and 10 display the output using OFDM and ATSE, respectively.



```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
HLPSL:
role reader(T, R: agent,
               Pur, Put: public_key,
               Prr: private_key,
               h: hash_funtion,
               FP: galos_field_number,
               P: basePoint(FP),
               Snd, Rcv: channel (dy))
played_by R def=
  local State : nat,
        C1, C2, SK1r, SK2t, X: text
        r, R1, T1, RKag: number
  init State := 0
```

Fig. 8 Entered codes into the AVISPA software



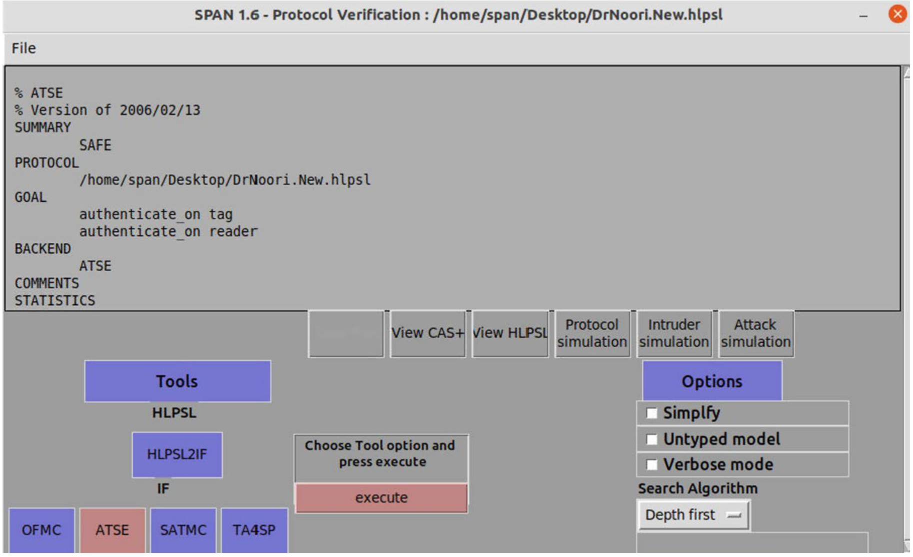
```
% OFMC
% Version of 2006/02/13
SUMMARY
        SAFE
PROTOCOL
        /home/span/Desktop/DrNoori.New.hlpsl
GOAL
        authenticate_on tag
        authenticate_on reader
BACKEND
        OFMC
COMMENTS
STATISTICS
```

□ Session Compilaction
Defth :
Path :

Fig. 9 Output of the proposed protocol using OFDM

4 Conclusion

Using ECC in smart cards can be highly effective in maintaining information security in different networks. Medical networks are obvious examples in which patient information is of great importance. In this paper, we have presented a scheme based on the ECC, which has a lower computational and communication cost and also lower elliptic curve point multiplication running time compared to the other present protocols. Data storage cost has also been analyzed and examined in this proposed protocol. Security requirements such as mutual authentication, confidentiality, data integrity, availability, anonymity, forwards security, location privacy, avoiding forgery attack, avoiding replay attack, avoiding man-in-the-middle attack and avoiding DoS attack have been considered,



SPAN 1.6 - Protocol Verification : /home/span/Desktop/DrNoori.New.hlpsl

```
% ATSE
% Version of 2006/02/13
SUMMARY
        SAFE
PROTOCOL
        /home/span/Desktop/DrNoori.New.hlpsl
GOAL
        authenticate_on tag
        authenticate_on reader
BACKEND
        ATSE
COMMENTS
STATISTICS
```

Options
☐ Simplfy
☐ Untyped model
☐ Verbose mode
Search Algorithm
Depth first

Fig. 10 Output of the proposed protocol using ATSE

which provide a reasonable security for RFID authentication in networks based on IoT. AVISPA software was also used for security analysis of the proposed protocol, and a safe output was obtained by the software. For future work, a hardware implementation can also be done in order to evaluate precise security and computational cost of the proposed method.

Abbreviations

ECC	Elliptic curve cryptography
IoT	Internet of things

Author contributions

All authors read and approved the final manuscript.

Funding

Not applicable.

Availability of data and materials

Not applicable.

Declarations

Competing interests

The authors declare that they have no competing interests.

Received: 14 January 2020 Accepted: 5 July 2022
Published online: 23 July 2022

References

1. J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, Internet of things (IoT): a vision, architectural elements, and future directions. Futur. Gener. Comput. Syst. **29**(7), 1645–1660 (2013)
2. P.P. Ray, A survey on internet of things architectures. J. King Saud Univ. Comput. Inform. Sci. **30**(3), 291–319 (2018)
3. A.A. Alamr, F. Kausar, J. Kim, C. Seo, A secure ECC-based RFID mutual authentication protocol for internet of things. J. Supercomput. **74**(9), 4281–4294 (2018)
4. L. Atzori, A. Iera, G. Morabito, The internet of things: a survey. Comput. Netw. **54**(15), 2787–2805 (2010)
5. Z. Zhang, Q. Qi, An efficient RFID authentication protocol to enhance patient medication safety using elliptic curve cryptography. J. Med. Syst. **38**(5), 47 (2014)

6. H.-Y. Chien, Elliptic curve cryptography-based RFID authentication resisting active tracking. Wireless Pers. Commun. **94**(4), 2925–2936 (2017)
7. H. Shen, J. Shen, M.K. Khan, J.-H. Lee, Efficient RFID authentication using elliptic curve cryptography for the internet of things. Wireless Pers. Commun. **96**(4), 5253–5266 (2017)
8. M.S. Farash, O. Nawaz, K. Mahmood, S.A. Chaudhry, M.K. Khan, A provably secure RFID authentication protocol based on elliptic curve for healthcare environments. J. Med. Syst. **40**(7), 165 (2016)
9. F. Rahman, M.Z.A. Bhuiyan, S.I. Ahamed, A privacy preserving framework for RFID based healthcare systems. Futur. Gener. Comput. Syst. **72**, 339–352 (2017)
10. L. Gao, L. Zhang, M. Ma, Low cost RFID security protocol based on rabin symmetric encryption algorithm. Wireless Pers. Commun. **96**(1), 683–696 (2017)
11. Y.K. Lee, L. Batina, I. Verbauwhede, EC-RAC (ECDLP based randomized access control): Provably secure RFID authentication protocol, in *2008 IEEE international conference on RFID*, 2008, pp. 97–104: IEEE
12. S. Kavitha, P. Alphonse, Y.V. Reddy, An improved authentication and security on efficient generalized group key agreement using hyper elliptic curve based public key cryptography for IoT health care system. J. Med. Syst. **43**(8), 260 (2019)
13. N. Koblitz, Elliptic curve cryptosystems. Math. Comput. **48**(177), 203–209 (1987)
14. V.S. Miller, Use of elliptic curves in cryptography, in *Conference on the theory and application of cryptographic techniques*, (Springer, Berlin Heidelberg, 1985), pp. 417–426
15. P. Tuyls, L. Batina, RFID-tags for anti-counterfeiting, in *Cryptographers' Track at the RSA Conference*, (Springer, Berlin Heidelberg 2006), pp. 115–131
16. Y.-P. Liao, C.-M. Hsiao, A secure ECC-based RFID authentication scheme using hybrid protocols, in *Advances in Intelligent Systems and Applications-Volume 2*: (Springer, Berlin Heidelberg, 2013), pp. 1–13
17. X. Yang, X. Yi, Y. Zeng, I. Khalil, X. Huang, S. Nepal, An improved lightweight RFID authentication protocol for internet of things, in *International Conference on Web Information Systems Engineering*, (Springer, Cham, 2018), pp. 111–126
18. K. Kaur, N. Kumar, M. Singh, M.S. Obaidat, Lightweight authentication protocol for RFID-enabled systems based on ECC, in *2016 IEEE Global Communications Conference (GLOBECOM)*, 2016, pp. 1–6: IEEE
19. K. Sowjanya, M. Dasgupta, S. Ray, An elliptic curve cryptography based enhanced anonymous authentication protocol for wearable health monitoring systems. Int. J. Inf. Secur. **19**(1), 129–146 (2020)
20. X. Li, J. Peng, S. Kumari, F. Wu, M. Karuppiah, K.-K.R. Choo, An enhanced 1-round authentication protocol for wireless body area networks with user anonymity. Comput. Electr. Eng. **61**, 238–249 (2017)
21. T.K. Dang, C.D. Pham, T.L. Nguyen, A pragmatic elliptic curve cryptography-based extension for energy-efficient device-to-device communications in smart cities. Sustain. Cities Soc. **56**, 102097 (2020)
22. K.-H. Wang, C.-M. Chen, W. Fang, T.-Y. Wu, A secure authentication scheme for Internet of Things. Pervasive Mob. Comput. **42**, 15–26 (2017)
23. C.-T. Li, D.-H. Shih, C.-C. Wang, Cloud-assisted mutual authentication and privacy preservation protocol for telecare medical information systems. Comput. Methods Programs Biomed. **157**, 191–203 (2018)
24. V. Kumar, M. Ahmad, A. Kumari, A secure elliptic curve cryptography based mutual authentication protocol for cloud-assisted TMIS. Telematics Inform. **38**, 100–117 (2019)
25. D. Noori, H. Shakeri, M. Niazi Torshiz, Scalable, efficient, and secure RFID with elliptic curve cryptosystem for Internet of Things in healthcare environment. EURASIP J. Inform. Security (2020). https://doi.org/10.1186/s13635-020-00114-x
26. P. Chinnasamy, S. Padmavathi, R. Swathy, S. Rakesh, Efficient data security using hybrid cryptography on cloud computing, in *Inventive Communication and Computational Technologies*, (Springer, Singapore, 2021), pp. 537–547
27. S.S. Dhanda, B. Singh, P. Jindal, Lightweight cryptography: a solution to secure IoT. Wirel. Pers. Commun. (2020). https://doi.org/10.1007/s11277-020-07134-3
28. D. Johnson, A. Menezes, S. Vanstone, The elliptic curve digital signature algorithm (ECDSA). Int. J. Inf. Secur. **1**(1), 36–63 (2001)
29. E. Barker, W. Barker, W. Burr, W. Polk, M. Smid, Recommendation for key management part 1: General (revision 3). NIST Spec. Publ. **800**(57), 1–147 (2012)
30. Z. Zhao, A secure RFID authentication protocol for healthcare environments using elliptic curve cryptosystem. J. Med. Syst. **38**(5), 46 (2014)
31. M. Nikooghadam, A. Zakerolhosseini, M.E. Moghaddam, Efficient utilization of elliptic curve cryptosystem for hierarchical access control. J. Syst. Softw. **83**(10), 1917–1929 (2010)
32. N. Koblitz, A. Menezes, S. Vanstone, The state of elliptic curve cryptography. Des. Codes Crypt. **19**(2–3), 173–193 (2000)
33. P. Alexander, R. Baashirah, A. Abuzneid, Comparison and feasibility of various RFID authentication methods using ECC. Sensors **18**(9), 2902 (2018)

Publisher's Note

Davood Noori received the BSc degree from university of Khavaran, Iran, in 2011, MSc from the Imam Reza University, Iran, in 2013, PhD from the Islamic Azad University, Sabzevar Branch, Iran, in 2019. His research focuses on data security, cryptography and wireless sensor network and RFID security. His current research interest is security in Internet of things (IOT) and reconfigurable architectures for multipliers under Galois field GF(2 m).

Hassan Shakeri received his BSc and MSc degrees in computer engineering from Ferdowsi University, Mashhad, Iran, in 1995, and Sharif University of Technology, Tehran, Iran, in 1997, respectively. In 2014, he received his PhD degree in computer engineering from Ferdowsi University. Currently, he is with the Department of Computer Engineering, Islamic Azad University of Mashhad. His research interests include trust management, computer system security and text processing. He has published more than 60 papers in national and international conferences and journals.

Masood Niazi Torshiz received the BS and MS degrees in computer engineering from Ferdowsi University, Mashhad, Iran, in 1997 and 2000, respectively, and the PhD degree from Islamic Azad University-Science and Research branch, Tehran, Iran, in 2008. In 2001, he joined Islamic Azad University-Mashhad branch as a faculty member. Since 2016, he has been serving as the Head of Department of Computer Engineering, Islamic Azad University-Mashhad branch. His research interests include cloud computing, IoT and database security.