RFID Authentication Scheme based on Hyperelliptic Curve Signcryption

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

10-01-2021 / 13-01-2021

CITATION

Ali, Usman (2021): RFID Authentication Scheme based on Hyperelliptic Curve Signcryption. TechRxiv. Preprint. <https://doi.org/10.36227/techrxiv.13551311.v1>

DOI

[10.36227/techrxiv.13551311.v1](https://doi.org/10.36227/techrxiv.13551311.v1)

RFID Authentication Scheme based on Hyperelliptic Curve Signcryption

USMAN ALI^{1,2}, MOHD YAMANI IDNA IDRIS¹, MOHAMAD NIZAM AYUB¹, INSAF ULLAH³,
IHSAN ALI¹, TARAK NANDY¹, MUKTAR YAHUZA¹, NOUMAN KHAN¹

¹Department of Computer System & Technology, Faculty of Computer Science & Information Technology, University of Malaya, Malaysia

²Department of Computer Science, University of Swat, Saidu Sharif Swat, Pakistan.

³Department of Information Technology, Hazara University Mansehra, Dhodial 21120, Pakistan.

Corresponding authors: Usman Ali, Mohammad Yamani Idris, Mohammad Nizam Ayub
(e-mail: usmanali838@ieee.org, ihsanalichd@siswa.um.edu.my, yamai@um.edu.my)

This work was supported in part by the University of Malaya Impact Oriented Interdisciplinary Research Grant Under Grant IIRG003(A.B.C)-19IIS, and in part by the Ministry of Higher Education Malaysia Fundamental Research Grant Scheme (FRGS) Under Grant FP055-2019A.

ABSTRACT The implementation of efficient security mechanisms for Radio Frequency Identification (RFID) system has always been a continuous challenge due to its limited computing resources. Previously, hash-based, symmetric-key cryptography-based and elliptic curve cryptography based security protocols were proposed for RFID system. However, these protocols are not suitable because some of them failed to fulfil the RFID security requirements, and some of them produce high computational overhead. Recently researchers have focused on developing an efficient security mechanism based on hyperelliptic curve cryptography (HECC) which provides high security with 80 bits lower-key size. In this paper, we propose an efficient RFID authentication scheme based on hyperelliptic curve Signcryption. The proposed authentication scheme provides the required security features for the RFID system as well as security from potential attacks. We validated our proposed scheme's security by utilizing a widely used simulation tool, Automated Validation of Internet Security Protocols and Applications (AVISPA). Furthermore, the results reveal that the computational, communication and storage overheads of the proposed scheme is much less than the other recently proposed schemes. Compared to the most recently published work based on ECC Signcryption, our scheme is 70% efficient in terms of computational overhead, 42.7% efficient in terms of communication overhead, and 50% efficient in terms of storage overhead. Therefore, the proposed scheme is more efficient as compared to the recently published work in this domain. Hence, it is an attractive solution for resource-limited devices like RFID tags.

INDEX TERMS Hyperelliptic Curve Cryptography, RFID, Authentication Protocol, AVISPA

I. INTRODUCTION

THE rapidly evolving computing age has enabled all sorts of possibilities to automate processes and recognize items that have now become crucial components of computing due to the fact it saves time and produces minimal errors as they pave the way to substantial productivity benefits. As of now, bar codes, voice recognition, optical character recognition, smart cards, magnetic stripes, chip cards, biometrics, and RFID are among the multitude of technologies that have been developed to incorporate Automatic Identification and Data Capture (AIDC). RFID has been proved to be the most popular AIDC technology in recent years [1]. According to Jia et al. [2], the use of RFID has grown exponentially

with the development of IoT, as the core technology behind it. RFID is a wireless communication technology using radiofrequency electromagnetic signals to detect and identify objects bearing tags [3]. There are three main communication components in an RFID system, namely: server, tag (transponder) and reader (Interrogator), as shown in Figure 1a. In an RFID system, the basic communication session begins when the reader broadcasts radio waves to interrogate the tag and the tag responds to the reader's signal. Figure 1b shows RFID system schematic diagram. RFID tags are classified into three distinct types: active tags, semi-passive tags and passive tags as shown in Table 1. An active tag carries onboard power-source that keeps the tag active to

transfer its data to an even larger range while a semi-passive tag has a minuscule onboard power-source; however, it only activates the tag in the presence of a nearby RFID reader. On the other hand, a passive tag has no onboard power-source. It can obtain the power required for activating the tag from the nearby RFID reader.

In general, an RFID tag contains a unique ID number, location information, object description such as price, date, etc. The ID helps the server to identify the tag distinctively in the presence of many tags. When a tag-carrying object enters the RFID reader region, its data is collected wirelessly and sent to the server for storage and user application requirements.

Khattab et al. [4] have discovered that when compared to various parameters, RFID outperforms other AIDC techniques, including data density, machine readability, human readability, cost, reading speed, range, moisture effect, and distraction of sight. However, RFID systems have some limitations, including less storage capacity and low processing speed of the tag. With such inadequate computational resources, designing and implementing security schemes that provide security features has been very demanding. Furthermore, the information is being transmitted between tags and readers wirelessly and is susceptible to attacks by the eavesdropper and illegitimate reader due to insecure wireless channels.

Several authors have addressed security concerns and challenges for RFID. Guizani et al. [7] and Kannouf et al. [8] presented an overview of RFID system threats and attacks. Khattab et al. [9] carried out a detailed analysis of RFID system attacks. They categorized RFID attacks into particularly three types: physical attacks, device attacks, and channel attacks. To prevent physical attacks, it is possible to avoid tag alteration or tampering by establishing a protected zone around the device using the device's sealed tamper-resistant case. Additionally, by using spread spectrum technologies and antenna polarization, the jamming attack can be countered. Furthermore, using a strong cryptographic scheme, the device attacks and other channel attacks can be prevented. Consequently, the cryptographic scheme must fulfill an RFID system's security requirements such as authentication, confidentiality, non-repudiation, integrity, anonymity, forward security, availability, and scalability. The cryptographic schemes that are prospective candidates to protect any information system are Symmetric Key Cryptography (SKC) [10], [11], RSA based cryptography [12], [13], Elliptic Curve Cryptography (ECC) [14], [15], and Hyperelliptic curve cryptography (HECC) [16], [17]. A comparison of these cryptographic systems based on various aspects has been presented in Table 2.

It can be observed from Table 2 that SKC based schemes have a big issue with key distribution, while RSA based schemes have high computational cost due to modular exponential computation. ECC-based schemes perform better than the RSA, while HECC performs better than ECC by providing the same security features with less computation cost, communication overhead, and memory requirement.

HECC based schemes require less storage, smaller key size, quicker in key generation. They produce smaller ciphertext as compared to other PKC schemes. Due to these features, HECC is an attractive cryptographic scheme to offer security for systems having limited computational resources such as RFID. Zheng [18] introduces the concept of Signcryption, which combines encryption as well as authentication in a single logical step. Before the actual advent of Signcryption, the technique was to use encryption-then-signature to achieve secrecy and authentication. Zheng showed that Signcryption saves 50% of computing time and 85% of communication costs compared to the process of signature-then-encryption.

A. MOTIVATION AND CONTRIBUTION

Providing security in all fields of computing and communication has always been a priority. However, implementing efficient and appropriate RFID system security mechanisms has been a continuous challenge because of limited computing resources. RFID system requires a security mechanism, that minimizes computational, communication and storage overhead. Recently Singh et al. [19] suggested Elliptic Curve Signcryption based RFID authentication protocol. The security and efficacy of their proposed protocol are based on ECC. Even though ECC utilizes 160-bit small keys and fewer parameter sizes as compared to RSA however, 160-bit key size is still not well suited for resource-limited devices like RFID tags. ECC has higher computation overhead and generates excessive communication cost compared to HECC having 80-bits key size [20] that generates less communication and computational cost than ECC. Hence, their proposed protocol does not fulfill performance efficiency because their scheme generates excessive communication overhead computational overhead. Furthermore, there is no verification of the security of their proposed method by using any verification tools such as Scyther and AVISPA. It is crucial to design authentication protocol, to eliminate all the above limitations, and to fulfill the security requirements of resource-limited RFID systems. We describe our main contributions as following.

- We designed an RFID authentication scheme based on hyperelliptic curve Signcryption
- We have shown that our scheme provides the required security features such as: authentication, confidentiality, non-repudiation, integrity, anonymity, forward security, availability and scalability.
- We have shown that our scheme provides security against replay, man-in-the-middle (MiM), impersonation, cloning, location tracking, desynchronization, Denial of Service (DoS), and key compromise attacks.
- The results of proposed scheme confirm its efficiency in terms of Computational, Communication and Storage overhead.
- We validated the proposed scheme's security by utilizing a popular simulation tool, Automated Validation of Internet Security Protocols and Applications (AVISPA).

Figure 1: Architecture of RFID System**Table 1:** Classification of RFID tags [5], [6]

Specification	Active Tags	Semi Passive Tags	Passive Tags
Cost	High	High	Low
Size	Large	Large	Small
Storage	Read/write memory	Read/write memory	Read memory
Power	Integrated battery	Internal chip battery	Surrounding signal
Distance	1000 m	100 m	5 m
Lifespan	10 years	10 years	Unlimited
Tag Signal	High	High	Low
Required Signal	Low	Low	High
Application	Environmental and logistic	Real-time tracking	Identification

B. STRUCTURE OF THE PAPER

The rest of the paper is organized as follows. Section 2 overviews the related work. Section 3 discusses the preliminaries of the hyperelliptic curve cryptography. Section 4 explains the proposed authentication scheme. Section 5 demonstrates the proof of the correctness of the proposed protocol. Section 6 provides the security analysis. Section 7 shows the performance analysis. Section 8 provides the conclusion and finally, appendix A provides the AVISPA code and simulation result.

II. RELATED WORK

Due to the lower computational capability of RFID tags, protection and privacy have been the main concern for RFID systems. Over the years, several security solutions providing various security features have been suggested. However, a great deal of emphasis has been placed on the design of secure authentication schemes for RFID. Also, several ECC-based schemes have been suggested in recent years, as ECC-based solutions are comparatively better than RSA and other PKC schemes. Gódor et al. [21] suggested ECC-based RFID authentication, which provides confidentiality and authentication while providing resistance against replay attacks. They also measured the computational time of various operations in the protocol. However, several required security attributes were not enforced in this scheme, and it could

not deter DoS attacks. Lee et al. [22] revealed the problems of un-traceability and anti-counterfeiting in RFID systems. They proposed a protection framework that offered several security features but failed to meet mutual authentication, scalability, and resistance to desynchronization attacks. Liu et al. [23] proposed ECC-based RFID authentication, which reduces RFID tag computation cost and provides mutual authentication, confidentiality, and anonymity. The protocol was capable of defending against desynchronization attacks, counterfeit attacks, and replay attacks. Liao et al. [24] suggested an RFID authentication protocol based on an elliptic curve that utilizes secure challenge-response and ID-verifier messages to be transferred. However, their protocol suffers from a key compromise attack where the key contained in the tag can be recovered by an attacker. Zhao [25] proposed an improved authentication protocol that is safer and powerful than the Liao scheme. However, Farash [26] showed that the Zaho scheme unsuccessful in offering forward security. Chou [27] categorized the RFID authentication schemes as ultra-lightweight, lightweight, simple, and full-fledged. Chou stated that full-fledged authentication schemes are attractive because non-full-fledged authentication schemes are susceptible to tracking and desynchronization attacks. Chou also suggested an ECC-based authentication scheme and asserted that it also offers forward anonymity and scalability in addition to location privacy and mutual authentication, while

Table 2: Comparison of different cryptographic schemes.

Evaluation parameters	Cryptographic Primitives			
	SKC	RSA	ECC	HECC
Computational cost	low	very high	high	lower than ECC
Communication overhead	low	very high	low	lower than ECC
Order of no of key	$O(n^2)$	$O(n)$	$O(n)$	$O(n)$
Key distribution	big problem	simple	simple	simple
Key bits for same security level	80 bit	1024 bit	160 bit	80 bit
Speed of key generation	speedy	very slow	speedy	speedy than ECC
Complexity	$O(n)$	$O(n^3)$	$O(n^2)$	$O(n^2)$
Memory requirement	very less	very large	Less	less than ECC

also offering security against DoS, replay, and MiM attack. Farash [28] has shown that the Chou scheme failed to provide forward secrecy, confidentiality, and mutual authentication. Furthermore, Chou protocol was proven to have failed to protect against tracking attacks, cloning attacks, and impersonation and attacks. Farash also suggested an enhanced authentication scheme that can address impersonation, tracking attacks, MiM attacks, and offer mutual authentication, confidentiality, and forward secrecy. However, this protocol's total computational time takes an even greater amount than existing RFID authentication schemes. Zhang et al. [29] suggested an ECC-based scheme, which provides session initiation anonymity. Conversely, Lu et al. [30] revealed that Zhang et al. protocol failed to offer mutual authentication. Lu et al. also suggested an updated authentication protocol to resolve the security vulnerabilities of the Zhang scheme. Mehmood et al. [31] exposed that Lu et al protocol is unable to protect the user's identity and does not provide resistance against masquerade attack. Mehmood et al. also suggested an enhanced mutual authentication scheme that provides anonymity, forward confidentiality, mutual authentication, forward protection, and session key privacy. Additionally, their scheme provides security against masquerade, replay, and MiM attacks. Feng et al. [32] suggested an ECC-based RFID security scheme that provides resistance against Dos, tracking, impersonation, and replay attacks. They asserted that their proposed scheme requires less communication overhead, storage costs, and processing time. Chen et al. [33] analyzed several ECC-based full-fledged RFID authentication schemes. They pointed out that some of these schemes have privacy and security drawbacks, while some schemes produce high communication costs. Chen et al also suggested two authentication schemes and stated that their schemes are efficient and secure. Shen et al [34], revealed that the Chen et al. protocol is susceptible to spoofing attacks and replay attacks. Alamr et al. [35] proposed an RFID authentication scheme using ECC based Diffie-Hellman key exchange concept to compute the secret key. This key, in turn, is utilized for the messages to be encrypted. Their protocol provides mutual authentication, confidentiality, anonymity, privacy and offers

resistance to replay, impersonation, and MiM attacks. Qian et al. [36] suggested lightweight RFID security protocol using ECC encryption and simple operations such as bitwise AND, XOR, etc. This scheme decreases the tag computation cost, as it does not use the operation of the elliptic curve scalar multiplication. However, this scheme is restricted to providing confidentiality, authentication, and forward secrecy only. Zheng et al. [37] suggested ECC-based RFID authentication that provides privacy, forward security, scalability, anonymity, and mutual authentication. Their protocol also offers resistance against DoS attacks, internal attacks, and tracking attacks. Chiou et al. [38] proposed an authentication protocol. However, in this protocol, the tag must perform five elliptic curve scalar multiplication (ECSM) operations, which increase computational cost. Therefore, this scheme is computationally inefficient and not appropriate for RFID systems. Liu et al. [39] proposed a security scheme for mobile RFID systems and mentioned that their approach is more efficient and can withstand all known attacks. Conversely, by observing the authentication stage of the protocol, it can be determined that the tag must execute four ECSM operations, which increases the tag computation cost. Therefore, this scheme is computationally inefficient and not appropriate for RFID systems. A lightweight RFID protection protocol for medical privacy was proposed by Fan et al. [40] and stated that it ensures confidentiality and secure authentication. This scheme is based on a simple operation such as XOR operation, hash computation, displacement operation, and cross operation. However, Aghili et al. [41] carried out a thorough review of the Fan et al. scheme and exposed that it is susceptible to secret information disclosures and impersonation attacks. Dinarvand et al. [42] suggested an ECC-based RFID authentication scheme that achieves authentication, non-repudiation, confidentiality, integrity, anonymity, forward security, availability, and scalability. Furthermore, their protocol is secure against tracking, de-synchronization, server spoofing, and replay attacks. Recently Singh et al. [19] proposed an RFID authentication protocol based on Elliptic Curve Signcryption. They also demonstrated that the computation cost and communication overhead of their

proposed scheme is less than others. As the security and efficiency of the authentication, protocols described in the above literature are based on an elliptic curve, using 160-bits key size, which generates excessive communication overhead as compared to the hyperelliptic curve with 80-bits lower-key size as shown in Table 2. Even though ECC utilizes 160-bit small keys and fewer parameter sizes as compared to RSA but still the 160-bit key size is not well suited for resource-limited devices like RFID tags.

III. PRELIMINARIES

The basic concept of hyperelliptic curve cryptography and its computational complexity is presented in this section.

A. HYPERELLIPTIC CURVE CRYPTOGRAPHY (HECC)

Hyperelliptic curves (HEC) are algebraic curves with genus $g > 1$ [16]. HEC are also known as generalized form of elliptic curves (EC) that have $g=1$. The difference between HECC and ECC is group operation. Unlike the EC, the points on the HEC cannot form a group; rather it generates an additive Abelian group, derived from the divisor class group. HEC of genus 2 with 80-bits field size can be constructed to attain similar security as 160-bits ECC [20]. A HEC of $g=2$ over $F(2^m)$, is a set of solution $(x, y) \in [F(2^m) \times F(2^m)]$ and is given by the equation (1):

$$E : y^2 + h(x)y = f(x) \quad (1)$$

where $[x, y] \in F(2^m)$, $h(x) \in F(2^m)$ $[x]$ is a polynomial with degree $\deg(h) \leq g$ and $f(x) \in F(2^m)$ $[x]$ is a monic polynomial of degree $\deg(f) = 2g+1$. Additional requirements for the curve is it must be non-singular curve. A divisor D as shown in equation (2), is a finite formal sum of scalar multiples of points in curve E .

$$D = \sum (m_p [P]) \quad (2)$$

where $m_p \in \mathbb{Z}$, and $[P]$ represent points on the hyperelliptic curve E .

B. COMPUTATIONAL COMPLEXITY

Let F_q is a finite field of size q , where $q \approx 2^{80}$.

1) Hyperelliptic curve Discrete Logarithm Problem (HECDLP)

It is computationally hard to compute an integer $d \in \{1, q-1\}$ given D and $p = d.D$.

2) Hyperelliptic curve Diffie-Hellman Problem (HECDHP)

Consider $a, b \in \{1, q-1\}$, it is computationally hard to compute abD , given (D, aD, bD) .

3) Hyperelliptic curve Decision Diffie-Hellman Problem (HECDDHP)

Consider $a, b, c \in \{1, q-1\}$, it is infeasible to determine whether $c = a.b$, given $(D, a.D, b.D, c.D)$.

IV. PROPOSED AUTHENTICATION SCHEME

Our proposed scheme is based on HEC Signcryption and is consisted of three phases: Setup, Authentication, and Update. The flowchart of our proposed scheme is shown in Figure 2. It is assumed that the data transmission from reader to server and vice versa is secure due to wired channel, whereas the data transmission from reader to tag and vice versa is insecure due to insecure wireless channel.

Table 3 shows the symbols and notations used to describe the scheme. The scheme is applicable to passive tag, semi-active tag and active tag.

Table 3: Symbols and notations used in the protocol.

Notation	Description
E	Hyperelliptic curve: $y^2 + h(x)y = f(x)$
F_q	Finite prime field of size $q = 2^{80}$
F_q^*	Algebraic closure of F_q
x, y	curve parameters
D	Divisor of HEC
V_s	Server private key
V_t	Tag private key
P_s	Server public key
P_t	Tag public key
T_{id}	Tag unique identifier
T_{pn}	Tag unique pseudonym
K	Secret Key
E_k	Encryption using K
D_k	Decryption using K
$hash$	One-way hash function

A. SETUP PHASE

The server in the Setup phase, performs the following operations to select and assign initial values to system parameters.

- i Selects a hyperelliptic curve $E : y^2 + h(x)y = f(x)$ with the curve parameters $\{F_q, F_q^*, q, x, y, D\}$.
- ii Select a unique identifier T_{id} for each tag such as $T_{id} = i.D$, where $i \in \{1, q-1\}$ is a random integer
- iii Select a random integer $T_{pn} \in_R \{1, q-1\}$, as the unique pseudonym for each tag.
- iv Selects a one-way hash function
- v The server stores $\{T_{id}, T_{pn}\}$ for every tag in its database.
- vi The server also stores $\{T_{id}, T_{pn}\}$ and $\{F_q, F_q^*, q, x, y, D\}$, in the memory of each tag.

B. AUTHENTICATION PHASE

The tag and server simultaneously authenticate each other by using the concept of signcryption-unsigncryption, in which authentication and confidentiality attributes are implemented together. The tag performs the signcryption operation while the server performs the unsigncryption operation. The

Figure 2: Flowchat of the proposed scheme.

complete work flow of the protocol has been presented in Figure 3. The following steps carried out in this phase:

- 1) For every session, the server initializes its private key V_s with a random number $\in \{1, q-1\}$. The server then compute public key P_s as and shown in Eq (3) and send it to the tag:

$$P_s = V_s D \quad (3)$$

- 2) The tag after receiving $\{P_s\}$, performs the Signcryption operation to obtain its Signcryption parameters C, R and S as follows:

- i For every session the tag initialize its private key V_t with random integer $\in \{1, q-1\}$ and compute its public key P_t as:

$$P_t = V_t D \quad (4)$$

- ii The tag computes its secret key K as:

$$K = \text{hash}(V_t P_s) \quad (5)$$

- iii The tag encrypt its pseudonym to obtain the first Signcryption parameter C as:

$$C = E_k(T_{pn}) \quad (6)$$

- iv The tag apply hash function to the XOR of tag-

identifier and tag-pseudonym to obtain second Signcryption parameter R as:

$$R = \text{hash}(T_{pn} \oplus T_{id}) \quad (7)$$

- v After computing R, the tag can now obtain its third Signcryption parameter S as:

$$S = (V_t / (V_t + R)) \text{mod} q \quad (8)$$

- vi The tag then sends C, R, S and P_t to the server.

- 3) The server performs the unsigncryption operation after receiving the Signcryption parameters C, R, S and tag public key P_t .

- i Computes its secret key K' as:

$$K' = \text{hash}(S.V_s(P_t + R.D)) \quad (9)$$

- ii Decrypts C by using K' to obtain first unsigncrypted parameter T'_{pn} as:

$$T'_{pn} = D_{K'}(C) \quad (10)$$

- iii Search its database to find the corresponding tag identifier T_{id} and if it is not found then the session is terminated, otherwise the second unsigncrypted

parameter R' is computed as:

$$R' = \text{hash}(T'_{pn} \oplus T_{id}) \quad (11)$$

- iv If $R = R'$, then server authenticate tag successfully, If $R \neq R'$ then authentication failed and session is terminated.
- v After tag authentication, the server computes authentication message $\{T_s\}$ as:

$$T_s = E_{k'}(R' \oplus T_{id}) \quad (12)$$

and sends $\{T_s\}$ to the tag.

- 4) Once the tag receive the authentication message $\{T_s\}$, it computes T_t as:

$$T_t = E_k(R \oplus T_{id}) \quad (13)$$

If $T_s = T_t$, then server authentication by tag is successful and if $T_s \neq T_t$, then authentication is unsuccessful and session is dismissed.

Figure 3: Proposed RFID authentication protocol.

C. UPDATE PHASE

Upon successful mutual authentication of tag and server, both of them must update the value of T_{pn} , so that it can be protected from desynchronization attack and unauthorized usage. The tag updates the T_{pn} by performing the following operation:

$$T_{pn}^{new} = (K \oplus T_{id} \oplus T_{pn}) \quad (14)$$

The server updates the T_{pn} by performing the following operation:

$$T_{pn}^{new} = (K' \oplus T_{id} \oplus T_{pn}) \quad (15)$$

V. PROOF OF CORRECTNESS

The accuracy of our proposed scheme is based on the fact that the same secret key has been produced by both parties in authentication phase. According to Eq. (5), the tag computed its secret key as shown below.

$$K = \text{hash}(V_t P_s), \text{ where } P_s = V_s D$$

$$K = \text{hash}(V_t V_s D).$$

According to Eq. (9), the server computed its secret key as shown below.

$$K' = \text{hash}(S.V_s(P_t + R.D)), \text{ where } P_t = V_t D.$$

$$K' = \text{hash}(S.V_s(V_t D + R.D))$$

$$K' = \text{hash}(S.V_s.D(V_t + R)) \text{ where } S = (V_t / (V_t + R)) \bmod q$$

$$K' = \text{hash}(V_t / (V_t + R).V_s.D(V_t + R))$$

$$K' = \text{hash}(V_t.V_s.D) = K$$

Since $K = K'$, hence the proposed scheme correctness is verified.

VI. SECURITY EVALUATION

The following assumptions are considered while performing security analysis.

A – 1. The tag identifier T_{id} and tag pseudonym T_{pn} is only known to server and tag.

A – 2. For every session, fresh random values for V_s and V_t are selected by the server and the tag, respectively.

A – 3. The encryption algorithm E_K is secure enough that an adversary is incapable of decrypting ciphertext C .

A – 4. If $P_t = V_t D$ and $P_s = V_s D$, then the adversary cannot obtain V_t and V_s , given P_t and P_s , due to HEC DLP. We followed the attack model of Ouafi and Phan [43], in which three queries are generated by Adversary A to model the passive and active attacks.

Query – 1: Execute (S, T, K) –In Passive attack; the adversary A, snoops the channel between tag T and server S, to retrieve session key k.

Query – 2: Send (S, T, K, M) –Adversary A impersonates server S and tag T in active attacks with some session key k, and sends its chosen message M to an instance of tag T or server S.

Query – 3: Corrupt (T, K) – Adversary A can obtain the secret K stored on the tag. This query can be omitted since there is no stored key K in tag memory in the proposed scheme.

A. SECURITY REQUIREMENTS

An overview of RFID security requirements achieved by the proposed scheme is discussed in this subsection. The required security attributes are authentication, confidentiality, non-repudiation, integrity, anonymity, forward security, availability, and scalability.

1) Authentication

In each session the tag should authenticate the server and vice versa, so that to ensure secure communication in RFID system.

- i Tag authentication: Once the server obtains the Signcryption text $\{C, R, S\}$ from the tag, it computes the key K' for decryption of ciphertext C to acquire the tag pseudonym T'_{pn} . The server searches its database to find the tag unique identifier T_{id} corresponding to T'_{pn} . The server computes R' and compare it with R obtained from the tag. If $R' = R$, then the server authenticates the tag successfully. Suppose an attacker pretends to be

a valid tag. In that case, it must produce an accurate value of R , but the value of R depends on T_{id} , and only legitimate tag and server know this, thus any illegitimate tag or reader cannot produce the correct value of R .

- ii Server authentication: The tag computes $\{T_t\}$, after it has received the message $\{T_s\}$ from the server. The message $\{T_s\}$ computed by the server depends on tag identifier T_{id} such that only legitimate tag and server know this. Additionally, $\{T_s\}$ is an encrypted message and an adversary is not able to find the shared secret key to decrypt the message $\{T_s\}$. The tag authenticates the server successfully if $T_s = T_t$. An adversary pretending itself to be a valid server must produce the correct message $\{T_s\}$. However $\{T_s\}$ depends on T_{id} , and only legitimate tag and server know this, thus it is impossible for an unauthorized server to produce the correct message $\{T_s\}$.

2) Confidentiality

Confidentiality is the assurance to keep the information secret during the transmission. In the proposed scheme, the first message is the server public key $\{P_s\}$ sent to the tag and since it is a public parameter and known to all, so it can be submitted as plaintext. The second message sent to the server is the Signcryption parameters $\{C, R, S\}$ and tag public key $\{P_t\}$. As $\{P_t\}$ is a public parameter and is known to all so it can be transmitted as plaintext. All the three Signcryption parameters $\{C, R, S\}$ reveals no information. The adversary is unable to decrypt the ciphertext C because it requires private key V_t of the tag to produce the secret key K . According to the property of HECDLP, an adversary cannot compute V_t , given P_t and D . Similarly, Adversary A cannot obtain any information from R and S because R is computed from one-way hash function and its reverse is impossible to compute and S is obtained using R . The third message sent to the tag is $\{T_s\}$ which is encrypted message and an adversary cannot obtain any information from this because it requires a secret key K and an adversary is unable to produce it due to the property of HECDLP. Therefore, confidentiality attributes are successfully provided by the proposed protocol.

3) Non-repudiation

The value of R and S , sent to the server by the tag depends on the tag identifier T_{id} . Similarly, the $\{T_s\}$ message sent to the server's tag also depends on the tag identifier T_{id} . Based on Assumption 1, if $R = R'$ then the tag would not repudiate that the message was sent by it to the server and if $T_s = T_t$, then the server would not repudiate that the message was sent by it to the tag.

4) Integrity

An adversary can't have two messages that have the identical message digest [37]. It means that for adversary A having an output of hash function can never determine the input message. Suppose an adversary alters any value in $\{C, R, S\}$, it can easily be identified by the server. The K value will not

be equal to the K value created by the tag, which causes the server to generate R incorrectly. In this situation, authentication fails and the server terminates the session. Similarly, it can be easily identified if an Adversary alters $\{T_s\}$ received by the tag from the server. It would not be the same as computed by the tag, in this situation, the authentication fails and the tag terminates the session. In the proposed protocol, data integrity during transmission is thus guaranteed.

5) Anonymity

In the proposed scheme, the tag sends its secret information: tag identity T_{id} and tag pseudonym T_{pn} in the form of a signcrypted message. Further, the security of T_{pn} is maintained by using encryption and the security of T_{id} is maintained by using a hash function to the result obtained from the XOR operations between T_{pn} and T_{id} . To obtain the confidential information T_{pn} and T_{id} , the adversary needs the secret key K , which is not possible for him to calculate due to HECDLP. Similarly, the secrecy of the tag identifier T_{id} is preserved in the message $\{T_s\}$ sent by the server to the tag by performing XOR operations between R' and T_{id} and then encrypting the result using the key k' . The adversary needs to decrypt to get the value of T_{id} , which is not possible because the key k' is not known and is not possible for him to calculate due to HECDLP. Therefore the suggested protocol offers tag anonymity.

6) Forward Security

Suppose the Adversary A, somehow manages to know the tag pseudonym T_{pn} . In that case, it may not be able to retrieve previous messages, due to the messages $\{T_s\}$ and $\{C, R, S\}$, solely dependent on the secret key computed by tag and server, which subsequently depends on the random numbers V_s and V_t generated by the server and the tag respectively for each session. The proposed scheme therefore offers forward security as the adversary yet is unable to get and use the past messages later.

7) Availability

The tag identifier T_{id} remains the same during the entire communication of tag and server, and the adversary is unable to reach it. In addition, the tag pseudonym T_{pn} that is sent to the server is updated for each session. The updating of tag pseudonym T_{pn} for both the server and the tag ensures that both have the same T_{pn} at all times. The proposed protocol, therefore, offers availability and prevents de-synchronization.

8) Scalability

The server searches and find tag identifier T_{id} in its database corresponding to the tag pseudonym T_{pn} obtained from the tag. So no linear search is needed for the server to know each tag's identity [28]. The server consumes $O(1)$ amount of time to search for the corresponding tag in the proposed scheme, consequently saving enormous computational workload as

total tags in the system increases. Thus, the proposed scheme, as a result, offers scalability.

B. ATTACKS

An RFID authentication scheme must be capable of safeguarding RFID system from attacks. This section analyzes various attacks on RFID system and the solution to counter these attacks.

1) Security against replay attack

An adversary eavesdropping on the communication channel can obtain the past messages $\{P_s\}$, $\{P_t, C, R, S\}$ and $\{T_s\}$, communicated between tag and server. The adversary then can replay these messages to create an unauthorized effect. In our proposed scheme, the tag pseudonym T_{pn} value is a private random number and for every new session, T_{pn} is updated to T_{pn}^{new} . Therefore in the new session the adversary is unable to use the previously recorded messages.

- i If an adversary pretending to be a valid tag and send to the server the pre-recorded message $\{P_t\}$ and $\{C, R, S\}$, then the server perform the following computation:

It computes the secret key as $K' = \text{hash}(S.V_s(P_t + R.D))$ and Decrypts the ciphertext to find the tag pseudonym $T'_{pn} = D_{K'}(C)$. The server is unable to find a corresponding tag identifier T_{id} because $T'_{pn} \neq T_{pn}^{new}$ and dismisses the session. The resistance of the server replay attack is shown in Figure 4a.

- ii If Adversary A pretending to be a valid server and transmit to the tag the pre-recorded messages $\{P_s\}$ and $\{T_s\}$, then the tag performs the following computations. Initialize its private key $V_t^{new} \in_R \{1 \dots q - 1\}$.

compute $P_t = V_t^{new} D$.

Compute its secret key $K^{new} = \text{hash}(V_t^{new} P_s)$.

Compute the ciphertext $C = E_{K^{new}}(T_{pn})$.

Compute $R^{new} = \text{hash}(T_{pn} \oplus T_{id})$.

Compute $S^{new} = (V_t / (R + V_t)) \text{mod} q$.

Compute $T_t^{new} = E_{K^{new}}(R \oplus T_{id})$.

Evidently, $T_t^{new} \neq T_s$ because in the new session the tag used V_t^{new} and the modified tag pseudonym T_{pn}^{new} , rather than V_t and T_{pn} , that were used in the former sessions. Thus, the server authentication fails and the tag terminates the session. The resistance of the tag replay attack is shown in Figure 4b.

2) Security against cloning attack

According to Liao et al. [24], the RFID authentication protocol is susceptible to cloning attacks when a group of tags uses the same secret key in the authentication process. The proposed authentication protocol uses no stored secret key in the tag memory. Each session's new key is produced dynamically therefore, the adversary is not capable to extract the confidential data to clone a tag. Even if the adversary can obtain a specific tag identifier T_{id} for a set of tags, it is not capable of obtaining tag pseudonym T_{pn} for the same tags, as $T_{id} \in_R 1 \dots q - 1$.

3) Security against location tracking attack

An adversary is unable to retrieve tag identifier T_{id} transferred between tag and server due to secure communication in our proposed scheme. Whenever the tag transmit $\{C, R, S\}$, the adversary would be unable to retrieve T_{id} , because it has to solve the computationally difficult HECDHP for computing the secret key. Likewise, the attacker can not decrypt when the server transmits $\{T_s\}$ to the tag due to the secret key K of the server and tag. Moreover, in producing the messages, private random numbers are used. The attacker will, therefore not access the location information, hence security against location tracking attack has been guaranteed.

4) Security against desynchronization attack

The adversary in the desynchronization attack prevents updating certain confidential information during tag server communication in an ongoing session. The adversary tries to intercept the messages and it is possible that the server failed to update the tag pseudonym in its database while the tag updates its pseudonym in its memory [42]. In our proposed scheme, the server stores the previous value of tag pseudonym T_{pn} as well as the modified value of tag pseudonym T_{pn}^{new} to avoid the desynchronization attack. When the server receives $\{C, R, S\}$ from the adversary, the server decrypts C and decides whether the decrypted value is a previous tag pseudonym T_{pn} or a modified tag pseudonym T_{pn}^{new} . But since the adversary has no correct values for T_{pn} and T_{pn}^{new} . Hence the adversary is unable to perform the de-synchronization of shared secret due to data integrity and mutual authentication provided by the proposed scheme.

5) Security against DoS attack

It has already been shown that while updating the tag pseudonym T_{pn} , the proposed scheme ensures availability and can also prevent de-synchronization attack. Furthermore, updating of the Tag Pseudonym T_{pn} between tag and server is the only synchronous update. Hence security against the DoS attack has been guaranteed.

6) Security against impersonation attack

An adversary eavesdropping on the communication channel can impersonate a valid server or tag.

- i Server impersonation attack is also known as server spoofing attack in which an adversary tries to mimic the behavior of a valid server. In doing so adversary A , chooses a random integer V_a , then calculates $P_a = V_a D$ and sends $\{P_a\}$ to the valid tag. The tag then produces the message $\{P_t, C, R, S\}$ and sends it to A . However, A is unable to find the correct tag identifier T_{id} corresponding to tag pseudonym T_{pn} and in turn, is unable to generate the message $\{T_s\}$ correctly and therefore unequal to the tag generated $\{T_t\}$. Thus, the tag finishes the session due to the fact $T_s \neq T_t$. The Adversary A therefore unsuccessful to mimic the behavior of a server and thus security against the server impersonation attack has been guaranteed as shown in Figure 5a.

Figure 4: Resistance against replay attack

Figure 5: Resistance against impersonation attack

ii The tag impersonation attack is also known as a tag masquerade attack in which an adversary tries to mimic the behavior of a valid tag. The adversary A, when receives $\{P_s\}$ from a valid server, generate the message $\{P_a\}$ and $\{C, R, S\}$ and send it to the server. Since the adversary is unable to obtain the tag identifier T_{id} , this is because only legitimate tag and server know it and therefore the message $\{C, R, S\}$ sent by the adversary is incorrect. The valid server when receiving this incorrect message $\{C, R, S\}$ from the adversary, decrypt the ciphertext C to compute R' , but since $R \neq R'$, hence the authentication fails and the session is terminated. Thus security against the tag impersonation attack has been guaranteed as shown in Figure 5b.

7) Security against MiM attack

An adversary in the MiM attack tries to modify the messages transmitted from the tag to the server and vice versa. The adversary pretending itself as a legitimate party and sends the modified messages to either tag or server[27]. As shown in section 6.2.6, the security against server and tag impersonation attacks is guaranteed and no illegitimate tag or

server initiates and completes the session successfully. Thus, security against the MiM attack has also been guaranteed.

8) Security against key compromise attack

Since the server and the tag randomly generate private keys V_s and V_t for each session, that are used to produce the secret key K and an adversary is unable to generate this secret key due to HECDLP. Hence security against the key compromise attack has been guaranteed.

VII. OBSERVATION AND ANALYSIS

A. FORMAL SECURITY EVALUATION

We implemented and validated the proposed scheme using the AVISPA simulation tool [44]. AVISPA is integrated with SPAN to provide a user interface. The architecture of the AVISPA tool has been shown in Figure 6. AVISPA tool operates under two validation states, namely: SAFE and UNSAFE. The output of the simulation is a SAFE state if a cryptographic scheme provides resistance against the MiM attack. The simulation's output is an UNSAFE state if a cryptographic scheme is not able to withstand the MiM attack. The role oriented language for writing cryptographic schemes in

Figure 6: Architecture of the AVISPA tool [44].

AVISPA is called High-Level Protocol Specification Language (HLPSSL) [45]. To simulate a cryptographic scheme to check whether it is SAFE or UNSAFE, AVISPA first converts the pseudo-code of the scheme into the HLPSSL source code. Thereafter, the HLPSSL2IF translator converts HLPSSL-code into an intermediate format (IF). The HLPSSL2IF translator validates the security of a protocol using four backend checkers called: Tree-Automata-based Protocol Analyser (TA4SP), SAT-based Model-Checker (SATMC), On-the-fly-Model-Checker (OF-MC), and CL-based Attack Searcher (CL-AtSe). According to [46], the HLPSSL2IF assesses a cryptographic scheme's security based on the initial knowledge of these backends. Furthermore, each backend has its functionality as discussed in [47], [48] and users can select any backend according to their needs. We used software tools such as SPAN (version: SPAN-Ubuntu-10.10-light_1) and Oracle VM Virtual Box (version: 5.2.0.118431). The HLPSSL source code of the proposed scheme contains four roles: role server, role tag, role session, and role environment as shown in Table 4, 5, 6, 7 respectively. AVISPA uses a special identifier i for the intruder. We used two backends of the AVISPA tool: OFMC and ASTE to validate our proposed scheme. The simulation result of the proposed protocol by using ATSE and OFMC back-end of AVISPA tool shows that the proposed protocol is safe as shown in Figure 7a and Figure 7b respectively.

B. OVERHEAD ANALYSIS

The proposed scheme's efficiency has been evaluated by measuring the common performance parameters that include computational, communication and storage overhead. This section provides the analysis of these overheads as well as the comparison of the results with the existing schemes.

1) Computational overhead

The computational overhead of an authentication scheme depends on the time consumed by various operations per-

Table 4: HLPSSL code for Server.

```

role Server(S,T:agent,
Ps,Pt:public_Key,Hash:hash_func,
SND,RCV:channel(dy))
played_by S def=
local
State: nat,
Na,Nal,Nb,Nb1,Tid,Tpn:text,
K1,R1:message
init
State := 0
transition
1.State =0/\RCV(start) =|>
State' :=1/\SND(S.T)
2.State =1/\RCV(T.{Nb'}_Ps) =|>
State' :=2/\Nal' :=new()
/\Na' :=Ps.Nal' /\SND(S.{Na'}_Pt)
/\K1' :=Hash(Na'.Nb')
/\secret(K1',sec_1,{S,T})
3.State =2/\RCV(T.{Tpn'.Tid'}_K1)=|>
State' :=3/\SND(S.{Tpn'}_K1)
/\witness(S,T,auth_1,Tpn')
/\R1' :=Hash(Tpn'.Tid')
/\request(S,T,auth_2,R1')
end role
  
```

Table 5: HLPSSL code for Tag

```

role Tag(S,T:agent,
Ps,Pt:public_key,Hash:hash_func,
SND,RCV:channel(dy))
played_by T def=
local
State: nat,
Na,Nal,Nb,Nb1,Tid,Tpn: text,
K1,R2: message
init
State := 0
transition
1.State =0/\RCV(S.T) =|>
State' :=1/\Nb1' :=new()
/\Nb' :=Pt.Nb1' /\SND(T.{Nb'}_Ps)
2.State =1/\RCV(S.{Na'}_Pt) =|>
State' :=2/\K1' :=Hash(Na'.Nb)
/\Tpn' :=new() /\Tid' :=new()
/\SND(S.{Tpn'.Tid'}_K1)
3.State =2/\RCV(T.{Tpn'}_K1) =|>
State' :=3/\request(S,T,auth_1,Tpn)
/\ R2' :=Hash(Tid.Tpn)
/\ secret(R2',sec_2,{S,T})
/\ witness(T,S,auth_2,R2')
end role
  
```

(a) Simulation results of OFMC.

(b) Simulation results of ATSE.

Figure 7: Simulation results

Table 6: HLPSSL code for Session role.

```

role session(S,T:agent,
Ps,Pt:public_key,
Hash:hash_func)
def=
local
SA,RA,SB,RB: channel(dy)
composition
  role_Server(S,T,Ps,Pt,
              Hash,SA,RA) /\
  role_Tag(S,T,Ps,Pt,
           Hash,SB,RB)
end role

```

Table 7: HLPSSL code for Environment role.

```

role environment()
def=
const
s,t: agent,
ka,kb: public_key,
h: hash_func,
sec_1,sec_2:protocol_id,
auth_1,auth_2:protocol_id
intruder_knowledge={s,t,h,ka,kb}
composition
session(s,t,ka,kb,h)
/\ session(s,i,ka,kb,h)
/\ session(i,t,ka,kb,h)
end role
goal
secrecy_of sec_1
  authentication_on auth_1
  secrecy_of sec_2
  authentication_on auth_2
end goal
environment()

```

formed by the protocol during its execution. In the ECC based RFID authentication protocol, the computational time is related to the number of elliptic curve scalar multiplication (ECSM) operation. Similarly, in HECC based RFID authentication protocol, the computational time is related to the number of hyperelliptic curve Division multiplication (HECDM) operations. The time consumed by other operations in an authentication scheme is very small compared to the execution time of ECSM or HECDM and therefore can be ignored. According to [21] the time to compute a single ECSM operation is $0.064s$ on a $5MHz$ tag. Thus, we can assume the time to compute a single HECDM to be $0.032s$ due to 80-bits key and parameter size which is half

of the key and parameters size used in 160-bits ECC [49]. In the proposed RFID authentication scheme, the tag executes one HECDM operation and the server executes two HECDM operations. Therefore, the tag execution time is $0.032s$ and the server execution time is $0.064s$. Therefore, the total time consumed by the server and tag together is $0.096s$. Table 8 and 9 compares the computation overhead with the current schemes [19], [24], [25], [29], [35], [37], [42]. Table 9 also provides the percentage improvement efficiency of the proposed scheme. A graphical representation of the comparison is also shown in Figure 8.

2) Communication overhead

Communication overhead depends on the size and the number of messages communicated between the two entities during the execution of a protocol. In the proposed scheme, three messages $\{P_s\}$, $\{P_t, C, R, S\}$, and $\{T_s\}$ are transferred between server and tag. We assumed, 128-bit AES for encryption that produces 128-bit ciphertext, while SHA-256 for a hash function, that produces 256 bits output. The communication cost of the tag to send the message $\{P_t, C, R, S\}$, is $128 + 256 + 80 + 80 = 544bits$. While the communication cost of the server to send the messages $\{P_s\}$ and $\{T_s\}$ is $80 + 128 = 208bits$. The total communication cost of tag and server is $752bits$. Table 10 presents a comparison of the communication overhead and improvement in efficiency from the current schemes [19], [24], [25], [29], [35], [37], [42]. A graphical analysis of this comparison is also shown in Figure 9.

3) Storage Overhead

The tag is required to store hyperelliptic curve parameters $\{F_q, F_q^*, q, x, y, D\}$, server public key P_s , tag's private key V_t , tag's public key P_t , tag's unique id T_{id} and the unique pseudonym of the tag T_{pn} and T_{pn}^{new} . Since 80-bit HECC has been used, the size of each curve parameter is 80 bits. So the storage cost of the tag can be calculated as: $80+80+80+80+80+80+80+80+80+80+80+80+80+80 = 960bits$. The server is required to store system parameters $\{F_q, F_q^*, q, x, y, D\}$, server private key V_s , tag unique identifier T_{id} , tag unique pseudonym T_{pn} , and T_{pn}^{new} . It is assumed that the system has m number of tags, so the storage cost of the server can be calculated as: $80+80+80+80+80+80+80+80+80m+80m+80m = 560 + 240mbits$. Table 11 presents comparison of the storage overhead and improvement in efficiency from the current schemes [20, 25, 26, 30, 36, 38, 43]. A graphical analysis of the comparison for the number of tags $m = 15$, is also shown in Figure 10.

C. COMPARISON OF SECURITY FUNCTIONALITIES

In this section the security requirements shown in section 6.1 and the potential to counter various attacks shown in section 6.2 are compared with the existing schemes [19], [24], [25], [29], [35], [37], [42] as shown in Table 12.

Table 8: Comparison of computational overhead.

Protocol	Tag	Server	Total
Singh et al. [19]	2 ECSM	3 ECSM	5 ECSM
Liao et al. [24]	5 ECSM	5 ECSM	10 ECSM
Zhao et al. [25]	5 ECSM	5 ECSM	10 ECSM
Zhang et al. [29]	4 ECSM	2 ECSM	6 ECSM
Alamar et al. [35]	4 ECSM	5 ECSM	9 ECSM
Zheng et al. [37]	3 ECSM	4 ECSM	7 ECSM
Dinarvand et al. [42]	3 ECSM	3 ECSM	6 ECSM
Proposed Protocol	1 HECDM	2 HECDM	3 HECDM

Table 9: Comparison of computational overhead

Protocol	Computational Overhead (sec)			proposed scheme efficiency
	Tag	Server	Total	
Singh et al. [19]	0.128	0.192	0.32	70%
Liao et al. [24]	0.32	0.32	0.64	85%
Zhao et al. [25]	0.32	0.32	0.64	85%
Zhang et al. [29]	0.256	0.128	0.384	75%
Alamar et al. [35]	0.256	0.32	0.576	83%
Zheng et al. [37]	0.192	0.256	0.448	78.6%
Dinarvand et al. [42]	0.192	0.192	0.384	75%
Proposed Protocol	0.032	0.064	0.096	—————

Table 10: Comparison of Communication overhead.

Protocol	Communication Overhead (bits)			Proposed scheme efficiency
	Tag	Server	Total	
Singh et al. [19]	736	576	1312	42.7%
Liao et. al. [24]	640	640	1280	41.2%
Zhao et. al. [25]	640	640	1280	41.2%
Zhang et. al. [29]	960	160	1120	32.8%
Alamr et. al. [35]	640	960	1600	53%
Zheng et. al. [37]	640	640	1280	41.2%
Dinarvand et.al. [42]	800	640	1440	47.8%
Proposed Protocol	544	208	752	—————

Table 11: Comparison of storage overhead.

Protocol	Storage overhead (bits)			Proposed scheme efficiency
	Tag	Server	Total	
Singh et al. [19]	1760	1120+640m	2880+640m	58.9%
Liao et. al. [24]	1920	1280+800m	3200+800m	66.3%
Zhao et. al. [25]	1760	1120+480m	2880+480m	49.2%
Zhang et. al. [29]	1600	1440+480m	2040+480m	44.6%
Alamr et. al. [35]	1920	1120+320m	3040+320m	34.7%
Zheng et. al. [37]	2080	1760+320m	3840+320m	40.7%
Dinarvand et.al. [42]	1920	1120+800m	3040+800m	65.9%
Proposed Protocol	960	560+240m	1520+240m	—————

Figure 8: Comparison of computational overhead.

Figure 9: Comparison of communication overhead.

Figure 10: Comparison of storage overhead.

Table 12: Comparison of security requirements and potential to counter various attacks.

Protocol	Security Attributes								Resistance against attacks								FSV
	MA	CO	NR	IN	AV	FS	AN	SC	LC	MiM	CL	RP	IM	DoS	DE	KC	
Singh et al. [19]	Y	Y	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	N
Liao et al. [24]	Y	Y	Y	Y	Y	Y	N	N	Y	Y	N	Y	Y	N	N	N	N
Zhao et al. [25]	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y	N	N
Zhang et al. [29]	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
Alamr et. al.[35]	Y	Y	Y	N	Y	Y	N	N	Y	Y	Y	Y	Y	N	N	Y	N
Zheng et. al. [37]	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
Dinarvand et.al. [42]	Y	Y	Y	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	N
Proposed scheme	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

MA-Mutual authentication, CO-Confidentiality, NR-Non Repudiation, IN-integrity, AN-Anonymity, AV-Availability, FS-Forward security, SC-Scalability, MiM-Man in the middle attack, CL-Cloning attack, RP-Replay attack, LC-location tracking attack, IM-Impersonation attack, DoS-Denial of service attack, DE-De-synchronization attack, KC-Key compromise attack, FSV-Formal Security Validation, N- Security functionality not satisfied. Y- Security functionality satisfied.

VIII. CONCLUSION

RFID technology has become very popular due to less expense and improved speed. However implementation of the security and privacy mechanism is a major problem for RFID tag due to its lower computational capacity. Previously, the researchers suggested hash-based, SKC-based, and ECC-based for RFID systems. However, some of these protocols failed to achieve complete security requirements and some protocols have high computational overhead. In this paper, we proposed a hyperelliptic curve Signcryption based RFID authentication scheme. The security and efficiency of the proposed scheme are based on 80-bit HEC as compared to 160-bit ECC. The proposed scheme achieves security requirements for the RFID systems such as authentication, confidentiality, non-repudiation, integrity, anonymity, forward security, availability, and scalability. Additionally, the proposed scheme can also provide security against replay, MiM, impersonation, cloning, location tracking, desynchronization, DoS, and key compromise attacks. Furthermore, the security of the proposed scheme is validated by using the AVISPA tool. The results of the performance parameters of the proposed scheme have been compared with seven recent RFID authentication protocols In terms of computation, communication, and storage overhead. Compared to the most recent protocol, our proposed scheme improves 70% computational overhead, 42.7% communication overhead, and 50% storage overhead. Thus the proposed scheme is more efficient and provides enhanced security as compared to the existing schemes, therefore, the proposed scheme is an attractive solution for resource-limited devices like RFID systems.

References

- [1] Roy Want. "An introduction to RFID technology". In: *IEEE pervasive computing 5.1* (2006), pp. 25–33.
- [2] Xiaolin Jia, Quanyuan Feng, Taihua Fan, et al. "RFID technology and its applications in Internet of Things (IoT)". In: *2012 2nd international conference on consumer electronics, communications and networks (CECNet)*. IEEE. 2012, pp. 1282–1285.
- [3] Jihong Yu and Lin Chen. *Tag Counting and Monitoring in Large-Scale RFID Systems*. Springer, 2019.
- [4] Ahmed Khattab, Zahra Jeddi, Esmail Amini, et al. "Introduction to RFID". In: *RFID Security*. Springer, 2017, pp. 3–26.
- [5] Claire Swedberg. "Iotera develops active RFID tag with 4-mile read range". In: *RFID Journal* (2014).
- [6] Paul N Roque. *Performance Analysis of Effective Range and Orientation of UHF Passive RFID*. Tech. rep. AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH GRADUATE SCHOOL OF ..., 2008.
- [7] Sghaier Guizani. "Security applications challenges of RFID technology and possible countermeasures". In: *2014 International Conference on Computing, Management and Telecommunications (ComManTel)*. IEEE. 2014, pp. 291–297.
- [8] Nabil Kannouf, Youssef Douzi, Mohamed Benabdelah, et al. "Security on RFID technology". In: *2015 International Conference on Cloud Technologies and Applications (CloudTech)*. IEEE. 2015, pp. 1–5.
- [9] Ahmed Khattab, Zahra Jeddi, Esmail Amini, et al. "RFID security threats and basic solutions". In: *RFID Security*. Springer, 2017, pp. 27–41.
- [10] Martin Ågren. *On some symmetric lightweight cryptographic designs*. Department of Electrical and Information Technology, Faculty of Engineering ..., 2012.
- [11] Hans Delfs, Helmut Knebl, and Helmut Knebl. *Introduction to cryptography*. Vol. 2. Springer, 2002.
- [12] Ronald L. Rivest, Adi Shamir, and Leonard Adleman. "A method for obtaining digital signatures and public-

- key cryptosystems". In: *Communications of the ACM* 26.1 (1983), pp. 96–99.
- [13] William P Wardlaw. "The RSA public key cryptosystem". In: *Coding Theory and Cryptography*. Springer, 2000, pp. 101–123.
- [14] Neal Koblitz. "Elliptic curve cryptosystems". In: *Mathematics of computation* 48.177 (1987), pp. 203–209.
- [15] Alfred J Menezes, Paul C Van Oorschot, and Scott A Vanstone. *Handbook of applied cryptography*. CRC press, 2018.
- [16] Neal Koblitz. "Hyperelliptic cryptosystems". In: *Journal of cryptology* 1.3 (1989), pp. 139–150.
- [17] Thomas Wollinger, Jan Pelzl, and Christof Paar. "Cantor versus Harley: optimization and analysis of explicit formulae for hyperelliptic curve cryptosystems". In: *IEEE Transactions on Computers* 54.7 (2005), pp. 861–872.
- [18] Yuliang Zheng. "Digital signcryption or how to achieve cost (signature & encryption) cost (signature)+ cost (encryption)". In: *Annual international cryptology conference*. Springer. 1997, pp. 165–179.
- [19] Anuj Kumar Singh and BDK Patro. "Elliptic Curve Signcryption Based Security Protocol for RFID." In: *KSII Transactions on Internet & Information Systems* 14.1 (2020).
- [20] Jan Pelzl, Thomas Wollinger, Jorge Guajardo, et al. "Hyperelliptic curve cryptosystems: Closing the performance gap to elliptic curves". In: *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer. 2003, pp. 351–365.
- [21] Győző Gódor, Norbert Giczi, and Sándor Imre. "Elliptic curve cryptography based mutual authentication protocol for low computational capacity RFID systems-performance analysis by simulations". In: *2010 IEEE International Conference on Wireless Communications, Networking and Information Security*. IEEE. 2010, pp. 650–657.
- [22] Yong Ki Lee, Lejla Batina, Dave Singelee, et al. "Anti-counterfeiting, untraceability and other security challenges for RFID systems: Public-key-based protocols and hardware". In: *Towards Hardware-Intrinsic Security*. Springer, 2010, pp. 237–257.
- [23] Ya-li Liu, Xiao-lin Qin, Chao Wang, et al. "A lightweight RFID authentication protocol based on elliptic curve cryptography". In: *Journal of computers* 8.11 (2013), pp. 2880–2887.
- [24] Yi-Pin Liao and Chih-Ming Hsiao. "A secure ECC-based RFID authentication scheme using hybrid protocols". In: *Advances in Intelligent Systems and Applications-Volume 2*. Springer, 2013, pp. 1–13.
- [25] Zhenguo Zhao. "A secure RFID authentication protocol for healthcare environments using elliptic curve cryptosystem". In: *Journal of medical systems* 38.5 (2014), p. 46.
- [26] Mohammad Sabzinejad Farash, Omer Nawaz, Khalid Mahmood, et al. "A provably secure RFID authentication protocol based on elliptic curve for healthcare environments". In: *Journal of medical systems* 40.7 (2016), p. 165.
- [27] Jue-Sam Chou. "An efficient mutual authentication RFID scheme based on elliptic curve cryptography". In: *The Journal of Supercomputing* 70.1 (2014), pp. 75–94.
- [28] Mohammad Sabzinejad Farash. "Cryptanalysis and improvement of an efficient mutual authentication RFID scheme based on elliptic curve cryptography". In: *The Journal of Supercomputing* 70.2 (2014), pp. 987–1001.
- [29] Zezhong Zhang, Qingqing Qi, Neeraj Kumar, et al. "A secure authentication scheme with anonymity for session initiation protocol using elliptic curve cryptography". In: *Multimedia Tools and Applications* 74.10 (2015), pp. 3477–3488.
- [30] Yanrong Lu, Lixiang Li, Haipeng Peng, et al. "A secure and efficient mutual authentication scheme for session initiation protocol". In: *Peer-to-Peer Networking and Applications* 9.2 (2016), pp. 449–459.
- [31] Zahid Mehmood, Gongliang Chen, Jianhua Li, et al. "A robust ECC based mutual authentication protocol with anonymity for session initiation protocol". In: *Plos one* 12.10 (2017), e0186044.
- [32] Li Feng and Xuanxia Yao. "RFID System Mutual Authentication Protocols Based on ECC". In: *2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom)*. IEEE. 2015, pp. 1644–1649.
- [33] Yalin Chen and Jue-Sam Chou. "ECC-based untraceable authentication for large-scale active-tag RFID systems". In: *Electronic Commerce Research* 15.1 (2015), pp. 97–120.
- [34] Han Shen, Jian Shen, Muhammad Khurram Khan, et al. "Efficient RFID authentication using elliptic curve cryptography for the internet of things". In: *Wireless Personal Communications* 96.4 (2017), pp. 5253–5266.
- [35] Amjad Ali Alamr, Firdous Kausar, and Jong Sung Kim. "Secure mutual authentication protocol for RFID based on elliptic curve cryptography". In: *2016 International Conference on Platform Technology and Service (PlatCon)*. IEEE. 2016, pp. 1–7.
- [36] Quan Qian, Yan-Long Jia, and Rui Zhang. "A Lightweight RFID Security Protocol Based on Elliptic Curve Cryptography." In: *IJ Network Security* 18.2 (2016), pp. 354–361.
- [37] Lijuan Zheng, Yujuan Xue, Linhao Zhang, et al. "Mutual Authentication Protocol for RFID based on ECC". In: *2017 IEEE International Conference on Computa-*

- tional Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC). Vol. 2. IEEE. 2017, pp. 320–323.
- [38] Shin-Yan Chiou, Wen-Tsai Ko, and Erl-Huei Lu. “A Secure ECC-based Mobile RFID Mutual Authentication Protocol and Its Application.” In: *IJ Network Security* 20.2 (2018), pp. 396–402.
- [39] Guoliang Liu, Huichuan Zhang, Fanyue Kong, et al. “A novel authentication management RFID protocol based on elliptic curve cryptography”. In: *Wireless Personal Communications* 101.3 (2018), pp. 1445–1455.
- [40] Kai Fan, Wei Jiang, Hui Li, et al. “Lightweight RFID protocol for medical privacy protection in IoT”. In: *IEEE Transactions on Industrial Informatics* 14.4 (2018), pp. 1656–1665.
- [41] Seyed Farhad Aghili and Hamid Mala. “Security analysis of an ultra-lightweight RFID authentication protocol for m-commerce”. In: *International Journal of Communication Systems* 32.3 (2019), e3837.
- [42] Negin Dinarvand and Hamid Barati. “An efficient and secure RFID authentication protocol using elliptic curve cryptography”. In: *Wireless Networks* 25.1 (2019), pp. 415–428.
- [43] Khaled Ouafi and Raphael C-W Phan. “Traceable privacy of recent provably-secure RFID protocols”. In: *International conference on applied cryptography and network security*. Springer. 2008, pp. 479–489.
- [44] Alessandro Armando, David Basin, Yohan Boichut, et al. “The AVISPA tool for the automated validation of internet security protocols and applications”. In: *International conference on computer aided verification*. Springer. 2005, pp. 281–285.
- [45] Yannick Chevalier, Luca Compagna, Jorge Cuellar, et al. “A high level protocol specification language for industrial security-sensitive protocols”. In: 2004.
- [46] Thomas Genet. “A short span+ avispa tutorial”. In: (2015).
- [47] Rifaqat Ali and Arup Kumar Pal. “Three-factor-based confidentiality-preserving remote user authentication scheme in multi-server environment”. In: *Arabian Journal for Science and Engineering* 42.8 (2017), pp. 3655–3672.
- [48] Jaewook Jung, Dongwoo Kang, Donghoon Lee, et al. “An improved and secure anonymous biometric-based user authentication with key agreement scheme for the integrated EPR information system”. In: *PloS one* 12.1 (2017), e0169414.
- [49] Insaf Ullah, Noor Ul Amin, Muhammad Naeem, et al. “A novel provable secured signcryption scheme????: A hyper-elliptic curve-based approach”. In: *Mathematics* 7.8 (2019), p. 686.

...