A Novel Hash-Based Mutual RFID Tag Authentication Protocol

Mansi Saxena, Rabindra Nath Shaw and Jitendra Kumar Verma

Abstract Radio frequency identification (RFID) is an integral part of our life. This term is coined for short-range radio communication technology. It is used to send and receive the digital information between stationary location and non-stationary object or between movable objects. It automates the wireless technology using radio waves to identify an object. This technology has widespread applications in the field of security, access control, transportation, etc. In this paper, we analyze an existing RFID-based protocol and demonstrate that it is insecure against impersonation attack, man-in-middle attack, server-masquerading attack, insider attack, and denial-of-service. We also propose a novel protocol, namely Encrypted Tag Identity and Secret Value Protocol, to overcome the shortcomings and loopholes existing in the surveyed protocols.

Keywords Impersonation attack · Server masquerading · Mutual authentication Public and private keys · Secret value · Random numbers

1 Introduction

In modern world, radio frequency identification (RFID) has become an important and integral part of human life for promoting productivity and convenience [1, 2]. RFID is a wireless technology that uses radio frequency electromagnetic fields to transfer data from a Tag attached to an object and thus provides non-contact system for the purpose

M. Saxena
AbyM Technology, Noida 201301, Uttar Pradesh, India

R. N. Shaw
School of Electrical Electronics & Communication Engineering,
Galgotias University, Greater Noida 201310, Uttar Pradesh, India

J. K. Verma (✉)
School of Computing Science & Engineering,
Galgotias University, Greater Noida 201310, Uttar Pradesh, India
e-mail: jitendra.verma.in@ieee.org

© Springer Nature Singapore Pte Ltd. 2019

L. C. Jain et al. (eds.), *Data and Communication Networks*, Advances in Intelligent
Systems and Computing 847, https://doi.org/10.1007/978-981-13-2254-9_1

of automatic identification and tracking of moving objects. It has multidimensional applications like transportation, automated payment system, object traceability, and access control. RFID refers to a technology where digital data encoded in RFID Tags are captured by a RFID Reader via radio waves. An RFID system contains three key components which are as follows.

(i) RFID Tags: It is attached to the object which has to be identified by the help of carrying identification data.
(ii) RFID Readers: It reads or writes the identification information on Tags using Radio waves.
(iii) Back-end database: It collect the records related to tagged objects and associate them with the information that is going to be read by the Reader.

RFID has drawn a significant attention in recent years. Many researchers have contributed a lot of research and products to this technology. In recent years, many RFID-based protocols are proposed for secure implementation and deployment of this technology. Srivastava et al. [3] proposed a secure and robust hash-based mutual RFID Tag authentication protocol in telecare medicine information system. They claimed that their protocol provides mutual authentication and is secure against eavesdropping and replay attacks. But after analyzing we demonstrate that the given protocol is vulnerable to impersonation attack, insider attack, server-masquerading and man-in-middle attack. Therefore, we introduce a new protocol, namely, Encrypted Tag Identity and Secret Value Protocol (ETISVP) which fills the loopholes of the existing protocol and proved to be safe against the impersonation, server-masquerading, denial-of-service and man-in-middle attacks.

Rest of the paper is organized as follows. Section 2 presents related work, Sect. 3 presents state-of-the-arts of hash-based Tag authentication protocol, Sect. 4 provides possible security threats from attacks and loopholes of the protocol under observation followed by proposed protocol and its security analysis along with proposed comparison of results of existing protocols and proposed protocol in Sects. 5 and 6. Section 7 concludes the paper.

2 Related Work

Most of the protocols developed for mutual authentication between the Tag and the Reader in RFID system are based on hash functions. Despite of being not very secure and containing loopholes, these protocols are easily exploited by the attackers. Many schemes have been proposed to counter the privacy issues in RFID System. Few of them are as follows.

Okubhu et al. [4] proposed a hash-based protocol, which was the improved version of "*Kill Command Feature*"-based protocols introduced by Auto-ID Center [5]. However, the high cost of searching tags remained the major flaw of this protocol. Weis et al. [6] proposed "*Hash Lock Technique*" which was the improved version of this protocol. Meanwhile, many protocols have been proposed such as randomized hash lock scheme by MIT, anonymous ID scheme by NTT, external re-encryption scheme by RSA Laboratory and XOR based onetime pad scheme by RSA Laboratory. Tsudik [7] proposed an improved authentication protocol, namely, Yet-Another Trivial RFID Authentication Protocol (YATRAP), in order to provide tracing resistance Tag authentication through monotonically increasing time-stamps on the Tag. Furthermore, they proposed a new protocol [8] which was found vulnerable to replay attack due to lack of Reader's authenticity.

Similarly, Chatmon et al. proposed an anonymous RFID authentication protocol in [9]. Sun et al. [10] proposed a RFID technology to guard inpatient medication safety. In 2009, Huang and Ku [11] proposed the RFID protocol for medication safety of inpatient, which was found to be vulnerable to denial-of-service and replay attacks.

Chien et al. [12] proposed improved version of [11] which was still remained vulnerable to the impersonation and replay attacks. Peris-Lopez et al. [13] introduced a new concept of IS-RFID system; however, it is vulnerable to easy manipulation [14–16]. Chen et al. [17] proposed tamper resistant protocol which fails to guarantee the safety against impersonation, desynchronization attacks and traceability. To overcome the desynchronization problem, Cho et al. [18] followed by Kim [15, 16] proposed hash-based RFID Tag mutual authentication protocols. Suja et al. proposed an RFID authentication protocol based on cyclic redundancy check (CRC) and hamming distance calculation between Reader to Tag. They claimed that their protocol resists against tracing and cloning attacks in the most efficient way [19].

Considering the great success of RFID technology in Telecare sector, Srivastava et al. [3] proposed a hash-based mutual RFID Tag authentication protocol in telecare medicine information system. We have reviewed and cross-examined their protocol against all possible attacks, and we have shown possible loopholes of this protocol in subsequent section (Table 1).

Table 1 Preliminaries

Symbol details	
I_k	Identity of the kth Tag
$\|$	Concatenation operation
N	Random number
N_r	Random number of Reader
N_t	Random number of Tag
S	Secret value
S_j	Secret value used in the jth session
$H(\cdot)$	Hash function
$\oplus$	Bitwise XOR
DB	Database

3 State-of-the-Arts

This section presents state-of-the-arts of hash-based mutual RFID Tag authentication protocol and its functioning. The underlying work is the basis of our proposed work in this paper. The hash-based Tag authentication protocol constitutes three phases that are as follows.

3.1 Pre-phase

Initially, the Tag and Data server share the Tag identity I_k, hash function $H(\cdot)$, and the secret value S_j. The Tag and the Reader have their own random number generators.

3.2 Reader's Request

The Reader generates the random number N_r and sends a request to Tag with this random number.

3.3 Tag's Response

On receiving request from the Reader, Tag is invoked and Tag generates a random number N_t and subsequently perform the following computations:

- $X = H(S_j \parallel I_k) \oplus N_\mathrm{t}$
- Calculate $Y = X \oplus H(I_k \parallel N_\mathrm{r} \parallel N_\mathrm{t})$, and then
- $Z = H(Y \oplus T_1 \oplus N_\mathrm{t})$.

Tag sends the response message $(X; Z; T_1)$ to the Reader and Reader sends the response message $(X; Z; T_1; N_\mathrm{r})$ to the Data server after adding N_r to it.

3.4 Data Servers Response and Tag Authentication

Data server after extracting the Tag identity (I_k) and shared Secret value (S_j) from the database performs the following computations:

- If the expected legitimate time interval for transmission delay, $\Delta T < (T_2 - T_1)$ the server rejects the login request, where T_2 is current time-stamp at server.
- Computes $N_\mathrm{t}^* = X \oplus H(S_j \parallel I_K)$.
- Check $Z^* = H(X \oplus H(I_k \parallel N_\mathrm{r} \parallel N_\mathrm{t}^*) \oplus N^* \oplus T_1) \approx Z$.

- Checks until Z^* is equal to Z as extracted from response message sent by Reader.
- Computes $W = H(X \oplus H(I_k \parallel N_\mathrm{r} \parallel N_\mathrm{t}) \oplus T_2 \oplus S_j)$.

When $Z^* \approx Z$, the Data server sends the Tag data to the Reader. It sends the message $U = \mathrm{DATA} \parallel W$ to the Reader. DATA is the information of Tag that needs to be transmitted.

3.5 On Receiving U, *Reader's Response*

On receiving U from Data server, Reader extracts the DATA from U and sends the remaining part to the Tag for further communication.

3.6 Data Server Authentication and Secret Value Updation

If $\Delta T < (T_3 - T_2)$, where T_3 is the current time-stamp and ΔT is the expected legitimate time interval for transmission delay then Tag rejects the request.

- Tag computes $W^* = H(Y \oplus S_j \oplus T_2)$.
- If $W^* \approx W$, Tag authenticates the Data server.
- Updates the S_j to $S_{j+1} = H(S_j \oplus N_\mathrm{r} \oplus N_\mathrm{t})$ on both Tag and server side.

4 Security Analysis against Possible Attacks and Loopholes

We perform security analysis of the protocol which emphasizes that the protocol is not secure and is vulnerable to various attacks such as any attacker can invoke the Tag by sending the random number N_r. The Tag performs following operations:

- $X = H(S_j \parallel I_k) \oplus N_\mathrm{t}$
- Calculates $Y = X \oplus H(I_k \parallel N_r \parallel N_\mathrm{t})$
- $Z = H(Y \oplus T_1 \oplus N_\mathrm{t})$.

Furthermore, the Tag sends the response message $(X; Z; T_1)$ to the Attacker assuming him to be the correct Reader. The attacker sends the response message $(X; Z; T_1; N_\mathrm{r})$ to the Data server adding N_r to it. Now with these values, the attacker can easily fetch the DATA from the message $U = \mathrm{DATA} \parallel W$ sent by the Data Server assuming the attacker to be the correct Reader.

4.1 Impersonation Attack

In the first step of protocol, the Reader sends the request to the Tag along with a random number N_{r}. At Tag's side, there is no verification mechanism to ensure that request is made by the correct Reader.

4.2 Server-Masquerading Attack

Since, the Tag identity (I_k) and the Secret value (S_j) are stored as plaintext in the database. Any privileged insider having access to database can easily get these values and behave as the Data server by intercepting the response message ($X; Z; T_1; N_{\mathrm{r}}$) send by the Reader. The Attacker with values ($I_k; S_j; X; Z; T_1; N_{\mathrm{r}}$) computes:

- $N_{\mathrm{t}}^* = X \oplus H(S_j \parallel I_k)$
- $\mathrm{W} = H(X \oplus H(I_k \parallel N_{\mathrm{r}} \parallel N_{\mathrm{t}}) \oplus T_2 \oplus S_j)$
- $U = \mathrm{InvalidDATA} \parallel W.$

In this way, the Attacker can send the invalid data to the Reader. The Reader will accept the message U from the Attacker, assuming him to be the Data server. Since, there is no mechanism for the verification of the correct Data server at the Reader's side.

4.3 Denial-of-Service Attack

The protocol is not fail-safe against the computation exhaustive attacks. At Data Server, there is an authentication step to verify the Reader.

- Check $\mathrm{Z}^* = H(X \oplus H(I_k \parallel N_{\mathrm{r}} \parallel N_{\mathrm{t}}^*) \oplus N_{\mathrm{t}}^* \oplus T_1) \approx Z.$

But there is no limit on the number of wrong messages can be sent to the Data server. The Attacker can send number of fake messages to the server which leads to the excessive computation on the server side keeping the server busy and unable to process any request.

4.4 Man-in-Middle Attack

All the messages between the Tag, Reader, and Data server are transmitted as plaintext on the communication channels. These messages can be intercepted and used by the Attacker as done in server-masquerading attack. Also the attacker can change these messages on the communication channel in order to invoke the denial-of-service attack by sending the fake messages to the Data server.

4.5 Insider Attack

Insider attacker is one who is having the administrative access to the server. Tag identity (I_k) and secret value (S_j) are stored as the plaintext to the server. The privileged insider, who has direct access to the server, can get these parameters and use the secret information for personal benefit as discussed in server-masquerading attack.

5 Encrypted Tag Identity and Secret Value Protocol: Proposed Work

We propose Encrypted Tag Identity and Secret Value Protocol (ETISVP) with the objective to fill the loopholes of the existing protocol. Our protocol uses the encryption scheme to safely communicate the messages between the Tag, Reader, and Data server. Encryption makes it impossible for the Attacker to intercept or modify the message during the transmission on communication channel. Table 1 shows the terminologies which are frequent in ETISVP.

The Tag identity (I_k) and Secret value (S_j) are stored as the hash function cipher in the database. So, the any privileged insider or attacker cannot hack and use these values from the database. The functioning of ETISVP is shown in Table 2.

5.1 Pre-phase

Initially, Tag, Reader, and Data server have their private keys for encryption and share their public keys with each other for decryption.

Hash function $H(\cdot)$, Tag identity (T) and secret value (V_j) are shared by Tag and Server. T and V_j are stored as H(T) and H(V_j) in the database. V_j is the secret value of Tag in jth session.

5.2 Readers Request

The Reader generates the random number N_r and performs the following computations.

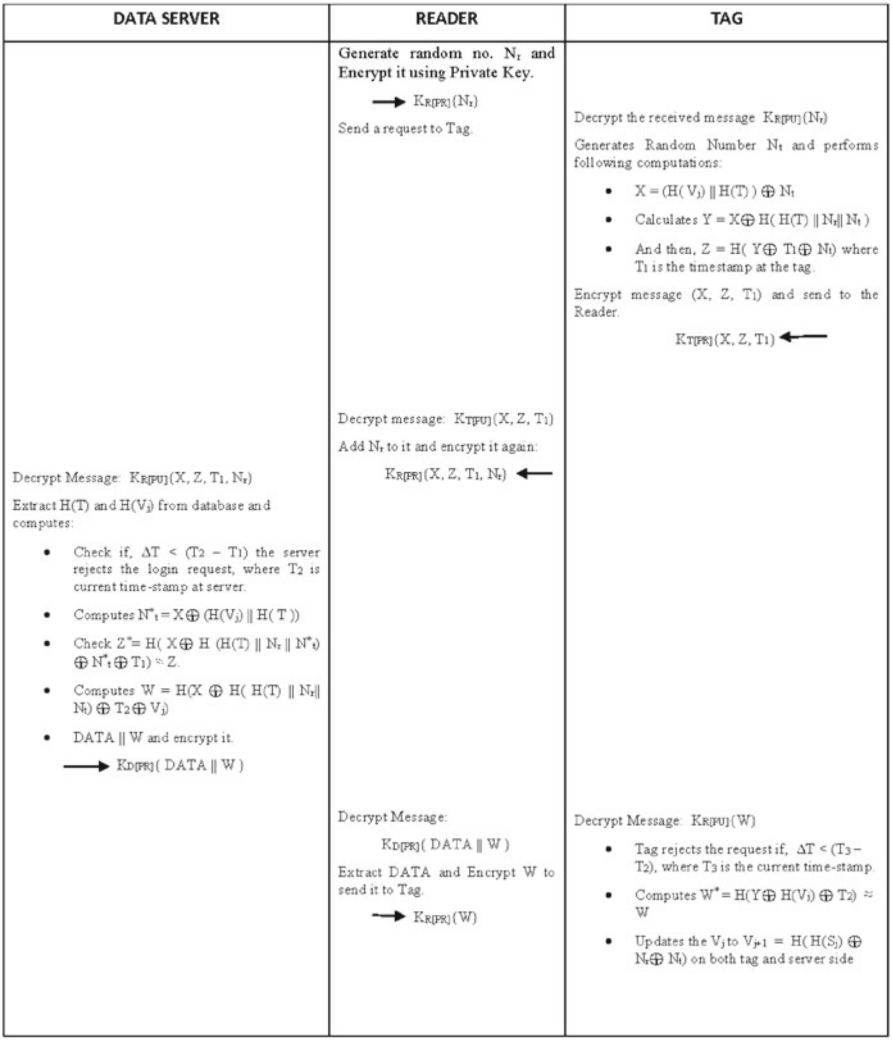
- Encrypt N_r using its private key, $K_{\mathrm{R[PR]}}\{N_\mathrm{r}\}$.
- Sends a request to Tag with $K_{\mathrm{R[PR]}}\{N_\mathrm{r}\}$ on secure communication channel.

5.3 Tags Response

Decrypt the message received from Reader using Reader's public key: $K_{\mathrm{R[PU]}}\{N_{\mathrm{r}}\}$. Generates a random number N_{t} and performs the following computations:

- $X = (H(V_j) \parallel H(T)) \oplus N_{\mathrm{t}}$
- Calculates $Y = X \oplus H(H(T) \parallel N_{\mathrm{r}} \parallel N_{\mathrm{t}})$, and then
- $\mathrm{Z} = H(Y \oplus T_1 \oplus N_{\mathrm{t}})$ where T_1 is the time-stamp at the Tag.

Table 2 Encrypted tag identity and secret value protocol



DATA SERVER	READER	TAG
	Generate random no. N_r and Encrypt it using Private Key. → $K_{R[PR]}(N_r)$ Send a request to Tag.	
		Decrypt the received message $K_{R[PU]}(N_r)$ Generates Random Number N_t and performs following computations: • $X = (H(V_j) \Vert H(T)) \oplus N_t$ • Calculates $Y = X \oplus H(H(T) \Vert N_r \Vert N_t)$ • And then, $Z = H(Y \oplus T_1 \oplus N_t)$ where T_1 is the timestamp at the tag. Encrypt message (X, Z, T_1) and send to the Reader. $K_{T[PR]}(X, Z, T_1)$ ←
	Decrypt message: $K_{T[PU]}(X, Z, T_1)$ Add N_r to it and encrypt it again: $K_{R[PR]}(X, Z, T_1, N_r)$ ←	
Decrypt Message: $K_{R[PU]}(X, Z, T_1, N_r)$ Extract $H(T)$ and $H(V_j)$ from database and computes: • Check if, $\Delta T < (T_2 - T_1)$ the server rejects the login request, where T_2 is current time-stamp at server. • Computes $N^*_t = X \oplus (H(V_j) \Vert H(T))$ • Check $Z^* = H(X \oplus H(H(T) \Vert N_r \Vert N^*_t) \oplus N^*_t \oplus T_1) \approx Z$. • Computes $W = H(X \oplus H(H(T) \Vert N_r \Vert N_t) \oplus T_2 \oplus V_j)$ • DATA $\Vert$ W and encrypt it. → $K_{D[PR]}(\mathrm{DATA} \Vert W)$		
	Decrypt Message: $K_{D[PR]}(\mathrm{DATA} \Vert W)$ Extract DATA and Encrypt W to send it to Tag. → $K_{R[PR]}(W)$	Decrypt Message: $K_{R[PU]}(W)$ • Tag rejects the request if, $\Delta T < (T_3 - T_2)$, where T_3 is the current time-stamp. • Computes $W^* = H(Y \oplus H(V_j) \oplus T_2) \approx W$ • Updates the V_j to $V_{j+1} = H(H(S_j) \oplus N_r \oplus N_t)$ on both tag and server side

Tag generates the response message (X, Z, T_1) and encrypts it using its private key: $K_{\mathrm{T[PR]}}\{X, Z, T_1\}$. Tag sends $K_{\mathrm{T[PR]}}\{X, Z, T_1\}$ to the Reader on secure communication channel.

5.4 *Reader's Side*

On receiving response of Tag on secure communication channel, the following operations take place on Reader's side.

- Decrypt the message received from Tag using Tag's public key: $K_{\mathrm{T[PU]}}\{X, Z, T_1\}$.
- Add N_{r} and encrypt the message using its private key: $K_{\mathrm{R[PR]}}\{X, Z, T_1, N_{\mathrm{r}}\}$.
- Reader sends $K_{\mathrm{R[PR]}}\{X, Z, T_1, N_{\mathrm{r}}\}$ to the Data server on secure communication channel.

5.5 *Data Server Response and Tag Authentication*

Decrypt message received from Reader using Reader's public key: $K_{\mathrm{R[PU]}}\{X, Z, T_1, N_{\mathrm{r}}\}$. Extracts $H(T)$ and $H(V_j)$ from database and performs the following computations:

If the expected legitimate time interval for transmission delay, $\Delta T < (T_2 - T_1)$

- The server rejects the login request, where T_2 is current time-stamp at server.
- Computes $N_{\mathrm{t}}^* = X \oplus (H(V_j) \parallel H(T))$.
- Check $Z^* = H(X \oplus H(H(T) \parallel N_{\mathrm{r}} \parallel N_{\mathrm{t}}^*) \oplus N_{\mathrm{t}} \oplus T_1) \approx Z$.
- Checks until Z^* is equal to Z.
- Computes $W = H(X \oplus H(H(T) \parallel N_{\mathrm{r}} \parallel N_{\mathrm{t}}) \oplus T_2 \oplus V_j)$
- If $Z^* \approx Z$ the Data server generates the message: DATA $\parallel W$.
- Encrypt it using its private key: $K_{\mathrm{D[PR]}}\{\mathrm{DATA} \parallel W\}$.

5.6 *Reader's Response*

On receiving the encrypted data from Data server, Reader performs the following operations.

- Decrypt the message using Server's public key: $K_{\mathrm{D[PU]}}\{\mathrm{DATA} \parallel W\}$.
- Extract DATA from $\{\mathrm{DATA} \parallel W\}$.
- Encrypt W using its private key: $K_{\mathrm{R[PR]}}\{W\}$.

And sends $K_{\mathrm{R[PR]}}\{W\}$ to the Tag on secure communication channel.

5.7 Data Server Authentication and Secret Value Updation

On receiving encrypted data from Tag, the following operations take place on Data server side.

- Decrypt the message using Reader's public key: $K_{\mathrm{R[PU]}}\{W\}$.
- Tag rejects the request if, $\Delta T < (T_3 T_2)$, where T_3 is the current time-stamp and ΔT is the expected legitimate time interval for transmission delay.
- Tag computes $W^* = H(Y \oplus H(V_j) \oplus T_2)$.
- If $W^* \approx W$, Tag authenticates the Data server.
- Updates the V_j to $V_{j+1} = H(H(S_j) \oplus N_{\mathrm{r}} \oplus N_{\mathrm{t}})$ on both Tag and server side.

6 Security Analysis

Proposed protocol ETISVP overcomes the loopholes of the existing protocol hash-based Tag authentication protocol, and hence, is highly safe against most common attacks like man-in-middle attack, eavesdropping attack, impersonation attack, server-masquerading attack and Insider Attack. The analysis of these attacks is provided in the following subsections.

6.1 Man-in-Middle Attack

In the proposed ETISVP protocol, during the message exchange, sender encrypts the message using the private key before transmitting it on communication channel. This message can only be decrypted by the corresponding public key. Therefore, any Attacker cannot modify the message without having the public key. Since, the attacker do not have the public key, he cannot use the message to harm the system.

6.2 Impersonation Attack

The Reader in the first step sends the random number to the Tag by encrypting it with the private key as $K_{\mathrm{R[PR]}}\{N_{\mathrm{r}}\}$. The Tag can decrypt the message by the corresponding public key. Therefore, the Attacker cannot send the random number to the Tag without knowing the private key of Reader to encrypt the message. Hence, the Attacker cannot fool the Tag by behaving as Reader.

Table 3 Performance Evaluation

Protocol	Computation cost (Tag)	Computation cost (Reader)	Communication rounds
YA-TRAP [13]	2H + 3RNG	RNG	4
Suja and Arivarasi [19]	3H + 2MOD + RNG	RNG	5
Existing Protocol [3]	2H + RNG	RNG	5
ETISVP	4H + RNG + Encryption	RNG + Encryption	5

6.3 Server-Masquerading Attack

According to the proposed protocol, Tag identity (T) and secret value (V_j) are stored in the encrypted form in database. Therefore, any privileged insider cannot fetch these values from database. Also, the message sent by the Reader to the Data server is encrypted by private key of Reader. Hence, cannot be decrypt by the Attacker without the corresponding public key. So the Attacker cannot get the required information for server masquerading.

6.4 Eavesdropping Attack

The proposed protocol is safe against the eavesdropping attack. Since, all the messages on the communication channel are in encrypted form. Therefore, any attacker cannot get the valuable information without knowing the corresponding public key for decryption. Hence, the proposed protocol meets the required challenges (Table 3).

7 Conclusion

In this paper, we have analyzed the hash-based mutual Tag authentication protocol based on random numbers and synchronized secret value. The protocol proved to be insecure against the eavesdropping attack, man-in-middle attack, impersonation attack, denial-of service attack, and insider attack. The protocol is vulnerable to these attacks due to the secret value, tag identity, and hash function stored in the database as plaintext. Also, the protocol requires safe communication mechanism for transferring intermediate computed values over the communication channel.

Proposed ETISVP protocol filled these loopholes by using encryption for message exchange and by storing the Tag identity and secret value in the database in encrypted form. The protocol introduced is secure against these attacks and hence, strengthen the security of RFID technology. The ETISVP provides immunity from several attacks; hence, it is safe to deploy in the industry. However, encryption of data delays the attack only rather than making it fool proof. In the modern world, we have several machines which are capable of holding huge computation power which poses threat to the proposed protocol for decrypting the encrypted information. Therefore, we are intended to work upon strengthening ETISVP as a future direction of work. Apart from this, we are also intended to work upon quantum computing environment as existing protocol are easily deciphered in quantum computing environment due to its immense computing power.

References

1. Shepard, S.: RFID: Radio Frequency Identification. McGraw Hill Professional (2005)
2. Landt, J.: The history of RFID. IEEE Potentials **24**(4), 8–11 (2005)
3. Srivastava, K., Awasthi, A.K., Kaul, S.D., Mittal, R.: A hash based mutual RFID tag authentication protocol in telecare medicine information system. J. Med. Syst. **39**(1), 153 (2015)
4. Ohkubo, M., Suzuki, K., Kinoshita, S., et al.: Cryptographic approach to privacy-friendly tags. In: RFID Privacy Workshop, Cambridge, USA, vol. 82 (2003)
5. Center, A.-I.: 860 MHz-960 MHz Class-I radio frequency identification tag radio frequency & logical communication interface specification proposed recommendation version 1.0. 0 (2002)
6. Weis, S.A., Sarma, S.E., Rivest, R.L., Engels, D.W.: Security and privacy aspects of low-cost radio frequency identification systems. In: Security in Pervasive Computing, pp. 201–212. Springer (2004)
7. Tsudik, G.: YA-TRAP: yet another trivial RFID authentication protocol. In: Pervasive Computing and Communications Workshops, 2006. PerCom Workshops 2006. Fourth Annual IEEE International Conference on, IEEE, pp. 4–7 (2006)
8. Tsudik, G.: A family of dunces: trivial RFID identification and authentication protocols. In: Privacy Enhancing Technologies, pp. 45–61. Springer (2007)
9. Chatmon, C., van Le, T., Burmester, M.: Secure anonymous RFID authentication protocols. Florida State University, Department of Computer Science, Tech. Rep
10. Sun, P.R., Wang, B.H., Wu, F.: A new method to guard inpatient medication safety by the implementation of RFID. J. Med. Syst. **32**(4), 327–332 (2008)
11. Huang, H.-H., Ku, C.-Y.: A RFID grouping proof protocol for medication safety of inpatient. J. Med. Syst. **33**(6), 467 (2009)
12. Chien, H.-Y., Yang, C.-C., Wu, T.-C., Lee, C.-F.: Two RFID-based solutions to enhance inpatient medication safety. J. Med. Syst. **35**(3), 369–375 (2011)
13. Peris-Lopez, P., Orfila, A., Mitrokotsa, A., Van der Lubbe, J.C.: A comprehensive RFID solution to enhance inpatient medication safety. Int. J. Med. Inform. **80**(1), 13–24 (2011)
14. Yen, Y.-C., Lo, N.-W., Wu, T.-C.: Two RFID-based solutions for secure inpatient medication administration. J. Med. Syst. **36**(5), 2769–2778 (2012)
15. Kim, H.: Enhanced hash-based RFID mutual authentication protocol. In: Computer Applications for Security, Control and System Engineering, pp. 70–77. Springer (2012)
16. Kim, H.: RFID mutual authentication protocol based on synchronized secret. Int. J. Secur. Appl. **7**(4), 37–50 (2013)
17. Chen, Y.-Y., Huang, D.-C., Tsai, M.-L., Jan, J.-K.: A design of tamper resistant prescription RFID access control system. J. Med. Syst. **36**(5), 2795–2801 (2012)
18. Cho, J.-S., Yeo, S.-S., Kim, S.K.: Securing against brute-force attack: a hash-based RFID mutual authentication protocol using a secret value. Comput. Commun. **34**(3), 391–397 (2011)
19. Suja, S., Arivarasi, A.: An RFID Authentication Protocol for Security and Privacy. In: International Conference on Computing and Control Engineering (2012)