International Journal of Advances in Electronics and Computer Science, ISSN(p): 2394-2835 Volume-6, Issue-9, Sep.-2019
http://iraj.in

SECURE AUTHENTICATED KEY AGREEMENT FOR TELECARE HEALTH SERVICES USING UBIQUITOUS IOT

[1]SUNGHYUN CHO, [2]HYUNSUNG KIM

[1,2]Dept. of Cyber Security, Kyungil University
[2]Dept. of Mathematical Sciences, University of Malawi, Malawi
E-mail: [1]chosh1110@naver.com, [2]kim@kiu.ac.kr

Abstract—The growth of wireless communication technologies and sensor technologies, telecare health services based on Internet things (IoT) are research focus from many researchers. However, security issue is the top most important focus to be solved for the success of IoT based telecare health service. This paper reviews Mahmood et al.'s authentication and prescription safety protocol and shows the security weaknesses in it, which is focused on denial of service attack and stolen-verifier attack. Furthermore, we propose a secure authenticated key agreement protocol for the safety of authentication and prescription for IoT based telecare health services. The proposed protocol efficiently solves the security problems in Mahmood et al.'s protocol.

Keywords—Internet of Things, Telecare Health Service, Prescription Safety, Authenticated Key Agreement, Authentication.

I. INTRODUCTION

The use of information and communication technology for telecare health services enables medical personnel and patients to perform the services via Internet of things (IoT) [1-3]. Thereby, there are tendencies that hospitals and healthcare organizations are adopting telecare medical information system (TMIS). TMIS can lower medical operating cost with improved quality of service and efficiency [4]. In spite of these advantages, several challenges should be addressed before TMIS can be adopted and deployed widely [5]. TMIS is vulnerable to many known security attacks, which are built on public networks. Medical history and private information of patents are maintained carefully by TMIS server and concealed in messages transmitted among entities to prevent user's privacy from being disclosed.

For the security and privacy concerns, there are many researches done on TMIS authentication and safe data transmission [6-15]. Wu et al. proposed two-factor authentication protocol for TMIS [6]. He et al. found Wu et al.'s protocol failed to resist insider attack and impersonation attack and proposed an improved protocol [7]. Wei et al. showed that both of Wu et al.'s protocol and He et al.'s protocol suffered from off-line dictionary attack and they designed their own protocol [8]. Zhu showed Wei et al.'s protocol still suffer from off-line dictionary attack [9].

Recent trends on three party password authentication key exchange (3-PAKE) protocol provide mutual authentication among patient, doctor and trusted server (TS) and hide identities from the adversary [10]. IoT can be an appropriate approach to support TMIS [11]. Rahimi et al. proposed a user authentication and key agreement for fitness-IoT structures [12-13]. Kim proposed a freshness preserving non-interactive hierarchical key agreement protocol, which is based on Bilinear pairing [14]. Recently, Mahmood et al. argued that the existing protocols lack in ensuring reliable prescription safety along with authentication for TMIS [15]. Furthermore, they proposed an authentication and prescription safety protocol for TMIS.

This paper reviews Mahmood et al.'s authentication and prescription safety protocol and shows the security weaknesses focused on denial of service attack and stolen-verifier attack. Furthermore, we propose a secure authenticated key agreement protocol for the safety of authentication and prescription for IoT based telecare health services. The proposed protocol efficiently solves the security problems in Mahmood et al.'s protocol.

II. REVIEW OF MAHMOOD ET AL.'S AUTHENTICATION AND PRESCRIPTION SAFETY PROTOCOL

This section reviews Mahmood et al.'s authentication and prescription safety protocol [15]. The purpose of their protocol is to protect patient's privacy and satisfy the security requirements of TMIS. There are four phases for the protocol between a new patient (A) to doctor/nurse (B) via TS.

A. Initialization by Patient

At the beginning, A chooses a random number $R_p$from a finite field and computes secret parameters. After that, X_A is calculated by multiplying random number $R_p$by an elliptic curve cryptosystem (ECC)-based generator P of large order n. Similarly, $Y_A$is the resultant of $R_p$and TS's public key F that is equal to dP, where d is random number a from finite field selected by TS. For level 1 encryption of security credentials, a hash of Y_A is taken to prepare key H_{YA}. A prepares a message M_A that contains hash of IDs and PW_pas A's password and message authentication code (MAC) is used for providing message integrity on TS side. A calculates $H(PW_p||ID_A||ID_B)$ and includes PW_p to keep it more secure. For transmission to the server, the patient computes cipher text P_A which is encrypted

by A's generated secret key H_{YA}as shown in step (iv). After that, a cipher text C_A is generated using a pre-established key K_{A-TS}. In step (vi), a temporary ID as $ID_{A\sim T}$of the patient is obtained by taking the hash of the $H(X_A, P_A, N_1)$ andN_1 is used for the current session only. The new $ID_{A\sim T}$ is never transmitted and can be calculated at TS using $H(X_A, P_A, N_1)$ where N_1 can be extracted after decryption. It encrypts the parameters $\{X_A, P_A, T_1\}$ using K_{A-TS} where, T_1 is timestamp. A transmits $\{ID_{A\sim T}, C_A\}$ to *TS* for authentication.

(i) $X_A = R_pP$
(ii)$Y_A = R_pF$
(iii) $M_A = H(PW_p||ID_A||ID_B)$
(iv) $P_A = E_{HYA}(ID_A||M_A||N_1||\text{MAC}(M_A)||ID_B)$
(v) $C_A = E_{KA-TS}(X_A||P_A||T1)$
(vi) $ID_{A\sim T}= \{H(X_A, P_A, N_1)\}$

B. Verification at TS

Upon receiving $\{ID_{A\sim T}, C_A\}$ from A, TS decrypts the cipher text C_A to get $(X_A||P_A||T_1)$. It also checks the message freshness by taking the difference from T_1 to guard against replay attacks. After that, TS computes the temporary key of the patient by multiplying the received X_A with d which was pre-generated by TS as Y_A'= dX_A. To verify whether the message is original, TS computes A's masked identity as $R_pF= R_pdP= dX_A$. It also decrypts P_A to obtain security credentials, including ID_A, M_A, N_1, MAC(M_A), and ID_B. The hash of these values is calculated as M_A'=$H(PW_p||ID_A||ID_B)$ and is then compared to verify the equality of $M_A$and M_A'to ensure message integrity. Otherwise, the message is discarded. MAC(M_A) provides data integrity for M_A. TS computes the following steps.

(i) Decrypt C_A using K_{A-TS} to get $\{(X_A||P_A||T_1)\}$
(ii) Computer Y_A'=dX_A
(iii) Decrypts P_A using $K_{H(YA)}$ to get $\{ID_A, M_A, N_1, \text{MAC}(M_A), ID_B\}$
(iv) Compute M_A'=$H(PW_p||ID_A||ID_B)$
(v) If verify ($MAC'(M_A)$!= MAC(M_A)) then discard
(vi) If M_A NOT equals M_A'then discard message

C. TS-based Mutual Authentication of B&A

After verification, TSpicks a random numberR_{Ts} andthen computes $Z_{TS} = H(ID_{TS}||ID_B||R_{Ts})$ using identities of B and TS. It also generates a nonce N_2 to get its hash with identities of communicating parties A and B. After that, TS calculates XOR of hash value with Z_{TS} to get a new temporary ID for B. The value of C_{TS} is obtained by encrypting $(ID_A||Z_{TS}||T_2||N_2)$ using the pre-established key K_{TS-B}. TS transmits the temporary identity $ID_{B\sim T}$and cipher text C_{TS} to B.

(i)$Z_{TS} = H(ID_{TS}||ID_B||R_{Ts})$
(ii) $ID_{B\sim T}= Z_{TS}XOR\ \{H(ID_B||ID_A||N_2)\}$
(iii)$C_{TS} = E_{KTS\text{-}B}(ID_A||Z_{TS}||T_2||N_2||ID_{B\sim T})$

$TS \rightarrow B : \{ID_{TS}, C_{TS}\}$

B receives the message $\{ID_{TS}, C_{TS}\}$ and decrypts it to get the other party's prescription details and TS validity by computing the set time stamp threshold value, nonce number, received masked-ID values, and decrypted message using the pre-share key from TS. At each end, entity $E_{KTS}$is used as a key to encrypt secure credentials in addition to MAC and the hash function application to make them more secure.

(i) Decrypt using K_{TS-B} to get $\{(ID_A||Z_{TS}||\ T_2||\ N_2)\}$
(ii) If $\{Z_{TS}\ XOR\ \{H(ID_B||ID_A||N_2)\}\}$ NOT equals $ID_{B\sim T}$ then discard
(iii) $X_B = R_BP$, $Y_B = R_BF$
(iv) $M_B = H(PW_B||ID_{TS}||ID_B)$
(v) $P_B = E_{HYB}(ID_B||M_B||N_3||\text{MAC}(M_B)||ID_{TS})$
(vi) $C_B = E_{KB-TS}(X_B||P_B||T_3)$

$B \rightarrow TS : \{ID_{B\sim T}, C_B\}$

TS receives the message $\{ID_{B\sim T}, C_B\}$ and decrypts it to get $(X_B||P_B||T_3)$. After that, TS computes Y_B'= dX_Bwhich is equal to $dR_BP= R_BdP= R_BF = Y_B$ calculated at B. It further decrypts P_B to get ID_B, M_B, N_3, MAC(M_B) and ID_{TS}, as illustrated in steps below. After that, TS verifies the message's integrity by computing and comparing the hash of the message. Finally, it computes the common parameters CP_A and CP_B for both parties and forwards them to A and B for session key computation.

(i) Decrypt C_B to get $[(X_B||P_B||T_3)]$
(ii) Computes Y_B'=dX_B
(iii) Decrypt P_B to get $[(ID_B||M_B||N_3||\text{MAC}(M_B)||ID_{TS})]$
(iv) Calculate M_B'= $H(PW_B||ID_{TS}||ID_B)$
(v) If M_B' NOT equals M_B then drop message
(vi) $CP_A = \{E_{HYA'}(X_B||ID_A||ID_B||Y_A'||N_1)\}$
(vii) $CP_B = \{E_{HYB'}(X_A||ID_A||ID_B||Y_B'||N_1)\}$

$TS \rightarrow A : \{ID_{A\sim T}, CP_A\}$

$TS \rightarrow B : \{ID_{B\sim T}, CP_B\}$

D. Participant Validation and Common Session Key Generation

A decrypts CP_A, verified by its own nonce and MAC which provide integrity and validity of TS and the message. The common parameters generated by TS are transmitted securely on each end. Upon receiving the secret credentials, the participating parties first verify message integrity and authority by verifying Y_A' and Y_B', respectively. After that, MAC, nonce, TS-ID, and the time stamp are also used for double-checking the source's integrity before processing secret credentials. After successful validation of both parties' identities and that of TS, participants start to compute the common key.

III. SECURITY WEAKNESSES ON MAHMOOD ET AL.'S PROTOCOL

This section shows two security weaknesses on Mahmood et al.'s protocol. They are denial of service attack and stolen-verifier attack.

E. Denial of service attack

Mahmood et al.'s protocol uses a temporary ID for the patient, which provides message freshness based on session different timestamp T_1. The usage of the temporary ID is for person anonymity of the patient, which claimed to be one of important factor of Mahmood et al.'s protocol.

TS should always reject any legal patient A's request in Mahmood et al.'s protocol. The reason is that TS requires to decipher C_A to get $(X_A||P_A||T_1)$ at verification phase. However, to decipher the cipher text, TS should choose any pre-share key with the targeting patient after identifying the patient based on $ID_{A\sim T}$. However, TS could not know the real patient with $ID_{A\sim T}$, which is obtained by taking the hash operation of X_A, $P_A$and N_1. Thereby, there is only possibility that TS rejects the legal patient's authentication requests, which results in denial of service.

The reason for the possibility of denial of service attack against Mahmood et al.'s protocol is that the protocol provides no method that TS could distinguish the message owner.

F. Stolen-verifier attack

Mahmood et al.'s protocol uses password to authenticate legal user and pre-shared secret key to provide secrecy of authentication and prescription safety. However, there are not small number of users for TMIS and thereby, Mahmood et al.'s protocol requires to use and keep verifier for the password and the pre-shared secret key.

The verification phase requires TS to decrypt C_A to get $(X_A||P_A||T_1)$ by using K_{A-TS}. Furthermore, the phase also requires TS to compute M_A'=$H(PW_p||ID_A||ID_B)$, which requires to know the passwordPW_p of the patient, for the validity verification to authenticate the patient.

The stolen-verifier attack means that an adversary who steals the password-verifier from the server can use it directly to masquerade as a legitimate user in a user authentication execution [16]. Note that the main purpose of an authentication protocol against the stolen-verifier attack is to reduce the immediate danger to user authentication. In fact, an adversary who has the verifier may further mount a guessing attack on it and success on Mahmood et al.'s protocol. The reason for the possibility of stolen-verifier attack against Mahmood et al.'s protocol is that the protocol needs to keep secret information in a verifier table so that TS could efficiently authenticate and keep prescription safety.

IV. SECURE AUTHENTICATED KEY AGREEMENT PROTOCOL

This sectionproposes a secure authenticated key agreement protocol for IoT based TMIS. The proposed protocol is consisted of four phases: setup phase, registration phase, login phase andauthenticated key agreement phase.

G. Setup Phase

TS performs system setup for the enhanced secure authenticated key agreement protocol. First of all, *TS* selects an elliptic curve *S* over E_q and a generator *P* of *S*, where *q* is a large order *n*. Also, *TS* selects a bilinear map $\hat{e}: G_1 \times G_1 \rightarrow G_2$. *TS* selects a secure one-way hash functions $h(\cdot)$: $\{0,1\}^* \rightarrow \{0,1\}^l$, where *l* is the length of output, selects its own random number *d* and computes its public key $F = \hat{e}(d, P)$. Finally, *TS* publishes $<E, P, F, h(\cdot), \hat{e}(\cdot)>$ as the system parameters.

H. PatientRegistration Phase

When a patient*A* wants to register with *TS*, this phase is necessary to be performed through a secure channel as follows.

Step 1: *A* selects his (or her) identity ID_A and sends it to *TS*.

Step 2: *TS* computes $V_A = h(ID_{TS}||ID_A||d)$ and issuesasmart card for *A*which stores $\{ E, P, F, h(\cdot), \hat{e}(\cdot), ID_{TS}, ID_B \}$.

Step 3: *A* computes $W_A = ID_A \oplus PW_A$, $V_1 = V_A \oplus W_A$ and $V_2 = h(W_A)$ by using his (or her) identity ID_A and password PW_A. After that, *A* deletes V_A from the memory of the smart card and writes $\{ V_1, V_2 \}$ on it.

I. Doctor/Nurse Registration Phase

Doctor/nurse*B* registration is the same as patient registration as follows

Step 1: *B*selects his (or her) identity ID_B and sends it to *TS*.

Step 2: *TS* computes $V_B = h(ID_{TS}||ID_B||d)$ andissuesasmart card for *B* which stores $\{ E, P, F, h(\cdot), \hat{e}(\cdot), ID_{TS}, ID_A \}$.

Step 3: *B* computes $W_B = ID_B \oplus PW_B$, $V_3 = V_B \oplus W_B$ and $V_4 = h(W_B)$ by using his (or her) identity ID_B and password PW_B. After that, *B* deletes $V_B$from the memory of the smart card and writes $\{ V_3, V_4 \}$ on it.

J. Login Phase

When *A* wants to communicate to*B*, *A* performs this login phase with *TS*. The details of this phase are as follows.

Step 1: *A* inputs ID_A and PW_A. *A*'s smart card computes $W_A' = ID_A \oplus PW_A$ and checks whether V_2equals to $h(W_A')$. If not, the smart card stops the phase.

Step 2: Otherwise, *A*'s smart card chooses a random number R_A and computes $X_A = \hat{e}(R_A, P)$, $Y_A = \hat{e}(R_A, F) \oplus ID_A$, $V_A' = V_1 \oplus W_A'$, $M_A = h(V_A||ID_A||ID_B)$ and $P_A = E_{K_A\text{-}TS}(M_A||\ ID_B)$. And then, sends the message $<X_A, Y_A, P_A>$ to *TS* through a public channel.

K. Authenticated Key Agreement Phase

Step 1: After *TS* receives the message $<X_A, Y_A, P_A>$, it computes $ID_A' = Y_A \oplus \hat{e}(d, X_A)$ and decrypts P_A using $K_{A\text{-}TS}$. Then, *TS* computes $M_A'=h(V_A||ID_A'||ID_B)$ and checks whether $h(V_A||ID_A'||ID_B)$ equals to M_A. If not, *TS* stops

the request. Otherwise, *TS* chooses a random number R_{TS} and computes$Z_{TS} = h(ID_{TS}||ID_B||R_{TS})$, $ID_{B\text{-}T}= Z_{TS} \oplus h(ID_B||ID_A)$ and$C_{TS} = E_{KB\text{-}TS}(ID_A||Z_{TS}||ID_{B\text{-}T})$. Then, *TS* sends the message $<X_A, C_{TS}>$ to *B*.

Step 2: Upon receiving$<X_A, C_{TS}>$ from*TS*, *B* decrypts C_{TS} using $K_{B\text{-}TS}$. After that, *B* computes $Z_{TS} \oplus \{h(ID_B||ID_A)\}$ and checks if $Z_{TS} \oplus h(ID_B||ID_A)$ equals to $ID_{B\text{-}T}$. If not, this session is aborted.

Step 3: Otherwise, *B* chooses a random number $R_B$andcomputes $X_B = \hat{e}(R_B, P)$, $Y_B= \hat{e}(R_B, F) \oplus ID_B$, $V_B' = V_3 \oplus W_B'$,$T_B = \hat{e}(R_B, X_A)$, $SK_B= h(T_B||ID_A||ID_B)$, $M_B= h(V_B'||ID_A||ID_B||K_{B\text{-}TS})$, $S= h(SK_B||ID_A||ID_B)$ and $P_B= E_{KB\text{-}TS}(M_B||S||\ ID_{TS})$ and sends the message $<X_B, Y_B, P_B>$ to *TS* through a public channel.

Step 4: After *TS* receives the message $<X_B, Y_B, P_B>$, it computes $ID_B'=Y_B \oplus \hat{e}(d, X_B)$ anddecryptsP_B using $K_{B\text{-}TS}$. Then, *TS* computes $M_B'=h(V_B||ID_B||ID_A||K_{B\text{-}TS})$ and checks whether $h(V_B||ID_B||ID_A||K_{B\text{-}TS})$ equals to M_B. If not, *TS* stops the request. Otherwise, *TS* computes $CP_A= E_{KA\text{-}TS}(X_B||S\ ||ID_A||ID_B)$ and forwards them to *A*for session key computation, respectively.

Step 5: Upon receiving $< CP_A>$ from *TS*, *A* decrypts CP_A using $K_{A\text{-}TS}$. After that, *A* computes $T_A = \hat{e}(R_A, X_B)$, $SK_A = h(T_A||ID_A||ID_B)$ and $S'=h(SK_A||ID_A||ID_B)$ and checks if S' equals to S.If not, the session is terminated. Otherwise, *A* and *B* could get the secure authenticated session key for telecare health service.

V. SECURITY ANALYSIS

This section provides security analysis focused on password guessing attack, replay attack, stolen-smart card attack and user anonymity.

L. Password Guessing Attack

Intheregistrationphase,the patient'spassword$PW_A$are used in $W_A$but not transmitted to *TS*. Although the privileged-insider of *TS* can obtain the registration message, it is unable them to know the registration entity's sensitive password because it isperformed only by the registration entity. Moreover, deriving the password from V_A stored in the smart card is equal to implementing the brute-force attack to crack the one-way hash function. Thereby, the proposed protocol is strong against password guessing attack.

M. Replay Attack

The usage of random numbers and timestamps is common solution for preventing replay attack in the authentication process. The messages $<X_A, Y_A, P_A>$, $<C_{TS}>$, $<X_B, Y_B, P_B>$, $<CP_A>$contain freshly generated random numbers in the proposed protocol. Furthermore, these random numbers are also embedded in the protected messages $X_A = \hat{e}(R_A, P)$, $Y_A = \hat{e}(R_A, F) \oplus ID_A$, $C_{TS}= E_{KB\text{-}TS}(ID_A||Z_{TS}||T_2||\ ID_{B\text{-}T})$. Thus, each participant needs to check the freshness of the message to cope from the replay attack. Hence, the proposed protocol discards the possibility of replay attack.

N. Stolen-Smart Card Attack

Suppose that an attacker get an smart card lost from a user and could read the stored parameters { *E*, *P*, *F*, $h(\cdot)$, $\hat{e}(\cdot)$ }. Then, the attacker could try to impersonate *A* or *B* to successfully login to *TS*. However, in the proposed protocol, the attacker cannot guess any candidate identity and password at the same time and compute V_1 andV_2. The way for the attacker to learn password is to find out the correct pair (ID_A, PW_A) such that $V_2 = h(W_A)$. In the proposed protocol, we assume the probability of guessing ID_A composed of exact *l* characters and PW_Acomposed of exact *m* characters is approximately $1/(2^{6l+6m})$. This probability is negligible, and the attacker has no feasible way to derive ID_A and PW_A in polynomial time. Thereby, the proposed protocol is safe from the stolen-smart card attack.

O. User Anonymity

Basedonthedesignoftheproposedprotocol,theexcellentpropertyofuseranonymitycanbeguaranteedateveryphase.Weusedmaskingfortherealidentity via a public channel, and no attacker can compromise user's real identity by launching security attacks. First, in the login phase, patient's real identity is included in $Y_A = \hat{e}(R_A, F) \oplus ID_A$.Thus, the attacker cannot reveal ID_A without using *d* to X_A due to elliptic curve discrete logarithm problem. Also, all of the identities are transmitted in cipher format instead of plaintext, and these identities will be randomized at each new session. As a result, the proposed protocol can provide user anonymity.

VI. PERFORMANCE ANALYSIS

This section provides performance analysis of the proposed protocol in terms of the computation complexities focused on the login phase and the authenticated key agreement phase only. We thus present a performance evaluation to compare the proposed protocol to Mahmood et al.'s protocol [15]. We present a comparison of the computational costs, and measure the execution time. The computational analysis of an authentication protocol is generally conducted by focusing on operations performed by each party within the protocols. Therefore, for analysis of the computational costs, we concentrated on the operations that are conducted by the parties in the network: namely a patient and a server. In order to facilitate the analysis of the computational costs, we define three notations, T_h, T_s and T_e, where $T_h$is for the time to execute a one-way hash operation, $T_s$is the time to compute a symmetric key encryption or decryption and $T_e$is for the time to compute an encryption or

decryption operation in ECC-160 algorithm.
In addition, in order to achieve accurate measurement, we performed an experiment. This experiment was performed using the Crypto++ Library[17] on a system using the 64-bits Windows 7 operating system, 3.2 GHz processor, 4 GB memory, Visual C++ 2013 Software, the SHA-1 hash function, the AES symmetric encryption/decryption and the ECC-160 operation. According to our experiment, $T_h$is nearly 0.0002 seconds on average, $T_s$is nearly 0.0087 seconds and $T_e$is nearly 0.6 seconds, respectively.

Overhead / Protocol	Patient	TS	D/N	Total
Mahmood et al[15]	$5T_h+3T_s+2T_e$	$6T_h+7T_s+2T_e$	$5T_h+4T_s+2T_e$	$16T_h+14T_s+6T_e$
The proposed	$3T_h+2T_s+2T_e$	$3T_h+4T_s+2T_e$	$3T_h+2T_s+3T_e$	$9T_h+8T_s+7T_e$

Table 1. Performance comparisons

Table 1 shows a comparative analysis of the computational cost among the related protocols. In addition, even though the proposed protocol has a bit of computational overhead than Mahmood et al.'s protocol,the proposed protocol assures higher security and privacy, and affords resistance to the most well-known attacks, while providing functionality.

VII. CONCLUSION

This paper first reviewed Mahmood et al.'s authentication prescription safety protocol and showed two security weaknesses. They were denial of service attack and stolen-verifier attack. Finally, we proposed a secure authenticated key agreement protocol for the safety of authentication and prescription for IoT based telecare health services. From the security analysis, we can argue that the proposed authenticated key agreement protocol efficiently solves the security problems in Mahmood et al.'s protocol.

ACKNOWLEDGMENTS

This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2017R1D1A1B04032598). Hyunsung Kim is the corresponding author.

REFERENCES

[1] S. J. Wu, R. D. Chiang, S. H. Chang, W. T. Chang, "An Interactive Telecare System Enhanced with IoT Technology," IEEE Pervasive Computing, vol. 16, pp. 62-69, 2017.
[2] R. Rao, "Internet of Things (IoT) Healthcare Benefits," The IOT Magazine, https://theiotmagazine.com/internet-of-things-iot-healthcare-benefits-2aae663c5c79, Jan. 8, 2018.
[3] H. Kim, E. K. Ryu, S. W. Lee, "Security considerations on cognitive radio based body area networks for u-healthcare," Journal of Security Engineering, vol. 10, no. 1, pp. 9-20, 2013.
[4] J. C. Tchatchoua, "Strategies for Improving Healthcare Efficiency While Reducing Costs," Walden University Doctoral Thesis, 2018.
[5] H. Wiong, J. Tao, C. Yuan, "Enabling telecare medical information systems with strong authentication and anonymity," IEEE Access, vol. 5, pp. 5648-5661, 2017.
[6] Z. Y. Wu, Y. C. Lee, F. Lai, H. C. Lee, Y. Chung, "A secure authentication scheme for telecare medicine information systems," Journal of Medical Systems, vol. 36, no. 3, pp. 1529-1535, 2012.
[7] D. He, J. Chen, R. Zhang, "A more secure authentication scheme for telecare medicine information systems," Journal of Medical Systems, vol. 36, no. 3, pp. 1989-1995, 2012.
[8] J. Wei, X. Hu, W. Liu, "An improved authentication scheme for telecare medicine information systems," Journal of Medical Systems, vol. 36, no. 6, pp. 3597-3604, 2012.
[9] Z. Zhu, "An efficient authentication scheme for telecare medicine information systems," Journal of Medical Systems, vol. 36, no. 6, pp. 3833-3838, 2012.
[10] D. Ku, H. Kim, "Enhanced User Authentication with Privacy for IoT-Based Medical Care System," International Journal of Computer Theory and Engineering, vol. 10, no. 4, pp. 125-129, 2018.
[11] H. H. Nguyen, F. Mirza, M. A. Naeem, M. Nguyen, "A review on IoT healthcare monitoring applications and a vision for transforming sensor data into real-time clinical feedback," in Proc. of 2017 IEEE 21st International Conference on Computer Supported Cooperative Work in Design, April 26-28, 2017.
[12] S. R. Moosavi, T. N. Gia, E. Nigussie, A. M. Rahmani, S. Virtanen, H. Tenhunen, J. Isoaho, "Session Resumption-Based End-to-End Security for Healthcare Internet-of-Things," in Proc. of 2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, pp. 581-588, 2015.
[13] S. R. Moosavi, T. N. Gia, A. M. Rahmani, E. Nigussie, S. Virtanen, J. Isaoaho, H. Tenhunen, "SEA: A secure and efficient authentication and authorization architecture for IoT-based healthcare using smart gateways," Procedia Computer Science, vol. 52, 452-459, 2015.
[14] H. Kim, "Freshness-Preserving Non-Interactive Hierarchical Key Agreement Protocol over WHMS," Sensors, vol. 14, pp. 23742-23757, 2014.
[15] Z. Mahmood, H. Ning, A. Ullah, X. Yao, "Secure Authentication and Prescription Safety Protocol for Telecare Health Services Using Ubiquitous IoT," Applied Sciences, vol. 7, Article no. 1069, 2017.
[16] S. W. Lee, H. Kim, K. Y. Yoo, "Cryptanalysis of a user authentication scheme using hash functions," ACM SIGOPS Operating Systems Review, vol. 38, no. 1, pp. 24-28, 2004.
[17] Dai, W. *Crypto++ Library 5.6.1* Available online: http://www.crpytopp.com(accessed on 2 Feb.2018).

★★★