View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Central Archive at the University of Reading

Full-Text version

Title: A Software Agent Enabled Biometric Security Algorithm for Secure File Access in Consumer Storage Devices

Authors: Ruhul Amin,
Department of Computer Science and Engineering, Thapar University,
Patiala, Punjab, India
(e-mail: amin_ruhul@live.com)

R. Simon Sherratt, *Fellow, IEEE*
Department of Biomedical Engineering, the University of Reading,
RG6 6AY, UK
(e-mail: sherratt@ieee.org)

Debasis Giri
Department of Computer Science and Engineering, Haldia Institute of Technology,
Haldia-721657, India
(e-mail: debasis_giri@hotmail.com)

SK Hafizul Islam
Department of Computer Science and Engineering, Indian Institute of Information Technology,
Kalyani, West Bengal 741235, India
(e-mail: hafi786@gmail.com)

Muhammad Khurram Khan, *Senior Member, IEEE*
Center of Excellence in Information Assurance (CoEIA), King Saud University,
Riyadh 11451, Saudi Arabia
(e-mail: mkhurram@ksu.edu.sa)

Publication: IEEE Transactions on Consumer Electronics
Publisher: IEEE
Volume: 63
Issue: 1
Date: February 2017
pp.: not yet assigned
DOI: not yet assigned

Abstract

In order to resist unauthorized access, consumer storage devices are typically protected using a low entropy password. However, storage devices are not fully protected against an adversary because the adversary can utilize an off-line dictionary attack to find the correct password and/or run an existing algorithm for resetting the existing password. In addition, a password protected device may also be stolen or misplaced allowing an adversary to easily retrieve all the stored confidential information from a removable storage device. In order to protect the consumer's confidential information that has been stored, this paper proposes a mutual authentication and key negotiation protocol that can be used to protect the confidential information in the device. The functionality of the protocol enables the storage device to be secure against relevant security attacks. A formal security analysis using Burrows-Abadi-Needham (BAN) logic is presented to verify the presented algorithm. In addition, a performance analysis of the proposed protocol reveals a significantly reduced communication overhead compared to the relevant literature.

Index Terms

Security Protocol, Biometric, Computer System, BAN logic, File Secrecy

I. INTRODUCTION

Consumer storage is commonly used to store and retrieve digital information. Consumers often store confidential information, files, or digital media purchases in the device. These devices are low cost and easily portable so the consumer often carries the device when travelling. As a result, the device may be lost or stolen by an adversary. If the confidential information is not protected, an adversary can easily retrieve the stored information from the device memory. However, the adversary faces a problem to retrieve the information from the store if the device is password protected. It is worth noting that a user's password (typically low entropy) cannot provide a strong secure system under a cryptographic dictionary attack. Indeed, many techniques are currently available to guess the password to access the device.

Mutual authentication and key agreement protocols are a popular paradigm in client-server environments to prevent unauthorized access. In 1981, Lamport [1] first introduced the secure communication client-server architecture and then numerous protocols [2]-[4] have been proposed for several applications, including wireless sensor networks [5], medical systems [6] and file security for USB based Mass Storage Devices (USB MSD) [7]-[12]. In order to provide secure access, authentication protocols play an important role.

Significant literature is now available to provide solutions to protect confidential files stored in a USB MSD. Yang *et al*. [7] first proposed a secure authentication protocol using the Schnorr Signature to protect the information stored. However, Chen *et al*. [8] argued that the protocol from Yang *et al*. [7] was not secure against the forgery attack and the replay attack. Furthermore, Lee *et al*. [9] argued that the protocol by Chen *et al*. [8] was computationally inefficient. In order to solve the security weaknesses, Lee *et al*. [9] proposed the three-factor authentication protocol based on elliptic curve cryptography. The protocol from Lee *et al*. [9] required the user's password, biometric and smartcard information as authentication tokens. More recently, He *et al*. [10] demonstrated that the protocol proposed by Lee *et al*. [9] was not secure against the password guessing attack, Denial-of-Service (DoS) attack and the replay attack, so proposed an improved three-factor authentication scheme. In order to resist the DoS attack, He *et al*. [10] employed the concept of the fuzzy extractor [13], [14]. In 2015, Amin and Biswas [15] proposed a three-factor authentication protocol for the same environment using a hash function to achieve a lower computation cost than existing protocols [9], [10].

This paper proposes a mutual authentication and key agreement protocol to provide only authorized access to confidential information stored on the device with the aid of a Registration Server (RS). A new user completes a registration procedure with RS allowing RS to deliver a link via e-mail from which the user can download and install registration software in their device which also incorporates the required secure access information relevant for only each user. In order to provide secure access to files, the user provides the necessary identity, password and biometric information. The device checks the legitimacy of the user and then negotiates a session key with RS. It is to be noted that this session key is used to encrypt the files in the storage device.

The rest of the paper is organized as follows: Section II presents an overview of the contribution and the novelty claims. Section III presents the hash function, fuzzy extractor and elliptic curve cryptography. The proposed protocol is provided in Section IV. The security analysis using BAN logic is discussed in Section V. Section VI provides the performance evaluation and comparison of the proposed protocol with related protocols. Section VII concludes the paper. TABLE I shows the nomenclature that is used throughout the paper.

TABLE I
NOMENCLATURE

Term	Usage
U_i	***i-th* user**
RS	Remote server
PW_i	Password of user U_i
BT_i	Biometric Template of user U_i
ID_i	Identity of user U_i
$E_k[]$	Symmetric key encryption using key k
$D_k[]$	Symmetric key decryption using key k
x	Secret key of the remote server
(P_x, P_y)	x and y coordinate of the elliptic curve point P
T_i	Current timestamp of U_i's storage device
T_j	Current timestamp of the Remote server
ΔT	Estimated time delay
UNSID	Unique software identity
SL	Software link
$h(\cdot)$	Cryptographic one-way hash function
REP()	*REP* procedure in fuzzy extractor
GEN()	*GEN* procedure in fuzzy extractor
$\oplus$	Bitwise XOR operator
$\|\|$	Concatenation operator
$(a.b)$	Point multiplication operation of a and b

II. SYSTEM ENVIRONMENT

In this work, a Registration Server (RS) delivers a link to all the users who have performed registration successfully, and then each user uses the link to obtain and install software in their device while also providing their credentials (password, identity and biometric signature.) Note that while the password may be guessed, it is hard to guess biometric signatures. Then, the software encrypts important files by using a negotiated key to provide security on the storage file. Whenever, the user of that device wants to access that file, RS first verifies the user and then provides a decryption key to recover the original file. All the files are then encrypted using a new session key. However, we argue that a storage device will still not be completely security protected. Hence, we have devised a standard security protocol which protects the storage device to defend unauthorized access. Firstly we have used the concept of biometric data along with a password in our protocol, hence it is difficult to guess the password along with biometric information. Secondly, an attacker cannot utilize a resetting technique, as we have mentioned in our protocol that if the attacker desires to use resetting technique, he/she first has to login into the system. As the attacker cannot login into the system without biometric data, the resetting technique is not usable.

This paper achieves the following contributions:

- ✓ A mutual authentication and key negotiation protocol to provide security protection of the stored information on the storage device,
- ✓ Security analysis to show that the proposed protocol is robust against known security attacks. Furthermore, in the proposed scheme, the mutual authentication and session key agreement have been verified using BAN logic.
- ✓ Significantly less communication overhead and computation costs than other related systems.

III. PRELIMINARIES

This section defines the fuzzy extractor [10]-[14] and the hash function [15] to analyze the security of the proposed protocol. Furthermore, the hardness assumption on the elliptic curve group is discussed.

Definition 1: A cryptographic one-way hash function maps a binary string of an arbitrary length to a binary string of fixed length, called the hashed value. It can be symbolized as: $h:\{0,1\}^* \rightarrow \{0,1\}^n$, where n is a positive integer. The properties of the hash function have been presented [4], [5].

Definition 2: A fuzzy system based collision resistant extractor can be modeled as a procedure which takes a binary string, say b, of some metric space $M \in \{0,1\}^n$ as an input for some positive number n and outputs a random string, say $\psi \in \{0,1\}^l$ for some

positive number *l* and an auxiliary string, say $\theta \in \{0,1\}^r$ for some positive number *r*, where *r* can be *l* or *n*. This mapping procedure is denoted by *GEN:* $M \to \psi \times \theta$. Another procedure which takes two inputs: (i) a binary string say, $b^{'}$ of the metric space $M \in \{0,1\}^n$, where $\left(b \neq b^{'}\right)$, and (ii) an uniform distribution binary string say, $\theta \in \{0,1\}^r$, and it produces the random string $\psi \in \{0,1\}^l$ as output. This mapping procedure is denoted by $REP: M \times \theta^{'} \to \psi$.

A. Elliptic Curve Cryptography (ECC)

The concept of elliptic curve cryptography was introduced by Kobiltz [16] and Miller [17], to design public key cryptosystems. Let $E_p(a,b)$ be a set of elliptic curve points over prime field F_p, where p is a large prime number. The elliptic curve equation is defined as: $y^2 = x^3 + ax + b \bmod p$ with $(a,b) \in F_p$ and $\left(4a^3 + 27b^2\right) \bmod p \neq 0$. The additive ECC group is defined as: $G_p = \{(x,y) : x, y \in F_p$ and $(x,y) \in E_p(a,b)\} \cup \{O\}$, where the point O is known as the 'Point at Infinity'. The scalar point multiplication on the cyclic group G_p is defined as: [*k*].*P* = *P* + *P* +… + *P*), that means *k* times addition of *P*.

Definition 3: Elliptic curve discrete logarithm problem: Given $\left(Q, R \in G_p\right)$, computation of the integer $k \in Z_p{*}$ is hard, where $R = [k].Q$.

Definition 4: Elliptic curve computational Diffie-Helman problem: Given $\left(P, [a].P, [b].P\right)$, for some $a, b \in Z_p{*}$, computation of $[a].[b].P$ is hard.

IV. PROPOSED PROTOCOL

This section describes the proposed mutual authentication and key negotiation protocol, which includes seven phases, (A) Registration and software installation phase, (2) Login phase, (C) Mutual authentication and key negotiation phase, (D) File management phase, (E) File accessing phase, (F) Password renewal phase and (G) Biometric renewal phase.

Initially, RS chooses a secret key x and computes $P_{pub} = [x].P$ as the corresponding public key. It should be noted that execution of the registration phase and the registration software installation phase is performed only once.

A. Registration and Software Installation Phase

Initially, each new user U_i must complete a registration procedure with RS. In this phase, U_i provides their information securely or in person (off-line mode) to RS. Then, RS securely sends to U_i, via e-mail, a link to downloadable registration software which must be installed in the storage device. The description of this phase is given below:

Step 1: U_i first chooses $\langle ID_i, PW_i \rangle$ and scans the user's biometric template, BT_i, such as a fingerprint. This work uses the biometric template to provide a high degree security since biometric templates cannot easily be forged [10]-[15]. U_i's device computes $PWB_i = h(PW_i \| b_i)$, where b_i is a random number generated by U_i and then sends $\langle ID_i, PWB_i, BT_i \rangle$ and a valid e-mail address to RS securely either using Transport Layer Security (TLS) or in person (off-line mode.)

Step 2: After receiving the registration message, RS computes $(\psi_i, \theta_i) = GEN(BT_i)$, $A_i = h(PWB_i \| \psi_i)$, $G_i = h(ID_i \| x)$, $B_i = G_i \oplus PWB_i$, $C_i = \theta_i \oplus h(ID_i \| PWB_i)$ and $D_i = E_{G_i}(A_i \| B_i \| C_i)$, where *GEN*() is the fuzzy extractor function.

Step 3: RS then embeds $\langle D_i, ID_i, B_i, GEN(), REP(), h() \rangle$ into the required registration software including all necessary parameters for the ECC cryptosystem. The registration software is a simple software application that must be installed in the consumer device. RS needs to maintain a database for storing all the registration information for all the consumers. RS stores

$(ID_i, UNSID_i, SL_i, \Xi)$ into the database, where $UNSID_i$ and SL_i are the unique software identity and software link respectively, and Ξ indicates empty attributes used to store the encrypted key. Finally, RS delivers to U_i via e-mail a link to user specific registration software (that includes SL_i.) This registration software is provided by the registration server to all the consumers with the software content varying with the user.

Step 4: After receiving the link for U_i to download the registration software, U_i installs it on their personal storage device. U_i then inputs b_i into the registration software. Finally, the registration software installed in U_i's storage device contains $\langle D_i, ID_i, B_i, b_i, GEN(), REP(), h() \rangle$.

B. Login Phase

This phase ensures that a non-registered user could not install the registration software without providing the correct information. The device runs the registration software now installed in the storage device and the software requests U_i to input their identity, password and biometric information (ID_i, PW_i and BT_i). Then the registration software checks the legitimacy of U_i by verifying the user's information by calculating $PWB_i^{'} = h(PW_i \| b_i)$, $G_i^{'} = B_i \oplus PWB_i^{'}$, $(A_i' \| B_i' \| C_i') = D_{G_i^{'}}(D_i)$, $\theta_i^{'} = h(ID_i \| PWB_i^{'}) \oplus C_i'$, $\psi_i' = REP(B_i, \theta_i^{'})$ and $A_i^{''} = h(PWB_i^{'} \| \psi_i^{'})$. The registration software checks whether the conditions $A_i^{''} = ? A_i'$ and $B_i' = ? B_i$ holds. If both the conditions are true, then the registration software of U_i accepts that the information provided by U_i is correct; otherwise, it aborts the session.

C. Mutual Authentication and Key Negotiation Phase

This phase first achieves mutual authentication and then negotiates a session key between the registration software of U_i and RS over an insecure channel. In this process, U_i and RS perform the following steps:

Step 1: U_i runs the registration software installed in his/her device and then provides their ID_i, PW_i and BT_i to the registration software. Then the registration software of U_i computes $PWB_i^{'} = h(PW_i \| b_i)$, $G_i^{'} = B_i \oplus PWB_i^{'}$, $(A_i' \| B_i' \| C_i') = D_{G_i^{'}}(D_i)$, $\theta_i^{'} = h(ID_i \| PWB_i^{'}) \oplus C_i'$, $\psi_i^{'} = REP(B_i, \theta_i^{'})$ and $A_i^{''} = h(PWB_i^{'} \| \psi_i^{'})$. The registration software in *Ui*'s device checks conditions $A_i^{''} = ? A_i'$ and $B_i' = ? B_i$. If both the conditions are not correct, registration software of U_i aborts the connection; otherwise, accepts U_i.

Step 2: The registration software in U_i generates random number r_i and sends $\langle ID_i, M_5, T_i \rangle$ to RS through an insecure channel, where $M_1 = [r_i].P$, $M_2 = [\theta_i].M_1$, $M_3 = (K_x, K_y) = [G_i^{'}].P_{pub}$, $M_4 = h(ID_i \| M_1 \| M_2 \| T_i \| K_y)$ and $M_5 = E_{K_x}(M_1 \| M_4 \| PWB_i \| C_i)$.

Step 3: After receiving $\langle ID_i, M_5, T_i \rangle$, RS first checks the existence of ID_i in the user database held by RS. If the entry does not exist then RS rejects the connection, otherwise RS checks the timestamp validity condition $|T_j - T_i| \leq \Delta T$ holds, where T_j is the current timestamp of RS. If it does not hold, RS rejects the connection; otherwise RS computes the legitimacy of U_i by computing $G_i^{'} = h(ID_i \| x)$,
$M_3^{'} = (K_x^{'}, K_y^{'}) = [G_i^{'}].P_{pub}$, $(M_1 \| M_4 \| PWB_i \| C_i) = D_{K_x^{'}}(M_5)$, $\theta_i^{'} = h(ID_i \| PWB_i) \oplus C_i$, $M_2^{'} = [\theta_i^{'}].M_1$ and
$M_3^{'} = h(ID_i \| M_1 \| M_2^{'} \| T_i \| K_y^{'})$. RS checks whether $M_3^{'} = ? M_3$ is true. If it is correct, then RS accepts U_i; otherwise, rejects U_i.

Step 4: RS generates random number r_j and computes $SK_j = [r_j].M_2^{'}$, $M_6 = h(ID_i \| PWB_i \| K_x^{'} \| r_j \| T_j)$ and $M_7 = E_{K_y^{'}}(M_6 \| r_j)$. RS sends M_7 to the registration software in U_i through a public channel.

Step 5: After receiving M_7, the registration software in U_i first checks whether the timestamp validity condition $|T_{jc} - T_j| \leq \Delta T$ holds, where T_{jc} is the current timestamp at the user end. If it fails, the registration software of U_i terminates the session; otherwise, it decrypts M_7 to obtain (M_6, r_j) as $(M_6 \| r_j) = D_{K_y'}(M_7)$. The registration software in U_i further computes $M_6' = h(ID_i \| PWB_i \| K_x' \| r_j \| T_j)$ and checks $M_6 = M_6'$. If true, RS is verified. Then registration software in U_i computes session key as $SK_i = [r_j].M_2$, which must be equal to SK_j and used to encrypt desired files stored in the memory of the consumer storage device.

D. File Management Phase

After performing mutual authentication and key negotiation, the registration software can encrypt any chosen files (F_1, F_2, ..., F_n), using the encryption key SK_i for security protection. Note that, the registration software in U_i can forget the encryption key after encrypting any files and send a confirmation message to RS. In this proposed protocol, RS maintains a table against each user U_i with the identity ID_i. Now, RS stores $(SK_i \oplus h(ID_i \oplus x))$ in the table against the identity ID_i.

E. File Accessing Phase

In this phase, U_i makes a request to RS to access the encrypted files stored in the consumer's storage device. In order to do it, U_i executes *Steps 1-3* of the mutual authentication and key negotiation phase to verify the legitimacy of U_i and generate a new session key. After the verification, RS first generates a random number $r_j' (r_j' \neq r_j)$ and then computes the new session key $SK_j' = [r_j'].M_2'$, where $(SK_j' \neq SK_j)$ and the random numbers are different in each session. Furthermore, RS then computes $M_6 = h(ID_i \| PWB_i \| K_x' \| r_j' \| T_j)$, $M_7 = E_{K_y'}(M_6 \| r_j')$ and retrieves $(SK_i \oplus h(ID_i \oplus x))$ from the local table in RS and then computes the old session key SK_i. Finally, RS computes $M_8 = E_{K_x'}(SK_i)$ and sends $\langle M_7, M_8 \rangle$ to U_i through an insecure channel. Then, the registration software in U_i decrypts M_7 and M_8 using K_y' and K_x' respectively. In order to verify the legitimacy of RS, the registration software in U_i computes $M_6' = h(ID_i \| PWB_i \| K_x' \| r_j' \| T_j)$. If $M_6' \neq M_6$, the registration software of U_i rejects the connection; otherwise, decrypts the encrypted files using the old key SK_i obtained from M_8 and can then access the files. After that, the registration software in U_i encrypts all the required files using the new key $SK_i' = SK_j' = [r_j'].M_2$. Finally, the registration software in U_i sends a confirmation message to RS that the obtained encrypted file is correct. Next, RS stores $(SK_i' \oplus h(ID_i \oplus x))$ in the table against ID_i.

F. Password Renewal Phase

This phase is infrequently used and the choice is dependent on the needs of the user. The description of the password update procedure is given as follows:

Step 1: U_i runs the registration software installed in their device, then provides their ID_i, the current PW_i and BT_i. Then the U_i registration software computes $PWB_i' = h(PW_i \| b_i)$, $G_i' = B_i \oplus PWB_i'$, $(A_i' \| B_i' \| C_i') = D_{G_i'}(D_i)$, $\theta_i' = h(ID_i \| PWB_i') \oplus C_i'$, $\psi_i' = REP(B_i, \theta_i')$ and $A_i'' = h(PWB_i' \| \psi_i')$. The registration software in U_i checks whether both $A_i'' = ? A_i'$ and $B_i' = ? B_i$ hold. If fasle U_i aborts the session.

Step 2: U_i inputs a new password PW_i^*. The registration software in U_i computes $PWB_i^* = h(PW_i^* \| b_i)$, $B_i^* = G_i' \oplus PWB_i^*$, $A_i^* = h(PWB_i^* \| \psi_i')$, $C_i^* = \theta_i' \oplus h(ID_i \| PWB_i^*)$ and $D_i^* = E_{G_i'}(A_i^* \| B_i^* \| C_i^*)$.

Step 3: Finally, the registration software in U_i replaces D_i with new value D_i^* and keeps the remaining information unchanged. Thus, U_i can change their old password without requesting any assistance from RS.

G. Biometric Renewal Phase

The execution of this phase is important whenever an existing user is willing to update their biometric information. The description of this phase is given as follows:

Step 1: U_i runs the registration software installed the device and then provides previous login information ID_i, PW_i and BT_i to the registration software. Then the registration software in U_i computes $PWB_i^{'} = h(PW_i \| b_i)$, $G_i^{'} = B_i \oplus PWB_i^{'}$, $(A_i^{'} \| B_i^{'} \| C_i^{'}) = D_{G_i^{'}}(D_i)$, $\theta_i^{'} = h(ID_i \| PWB_i^{'}) \oplus C_i^{'}$, $\psi_i^{'} = REP(B_i, \theta_i^{'})$ and $A_i^{''} = h(PWB_i^{'} \| \psi_i^{'})$. The registration software in U_i checks that both conditions $A_i^{''} = ? A_i^{'}$ and $B_i^{'} = ? B_i$. If false, the registration software in U_i aborts the session.

Step 2: U_i inputs new the biometric table BT_i^*. the registration software of U_i computes $(\psi_i^*, \theta_i^*) = GEN(BT_i^*)$, $A_i^* = h(PWB_i \| \psi_i^*)$, $C_i^* = \theta_i^* \oplus h(ID_i \| PWB_i)$, and $D_i^* = E_{G_i}(A_i^* \| B_i \| C_i^*)$.

Step 3: Finally, the registration software in U_i replaces D_i with the new value D_i^* and keeps the remaining information unchanged. Thus, U_i can change/renew biometric information without requesting any assistance from RS.

V. SECURITY ANALYSIS

This section explores the security of the proposed mutual authentication and key negotiation protocol. This work employs BAN logic [5], [10], [18], [19] to demonstrate that the proposed protocol provides secure authentication. The informal security analysis examines that the proposed protocol is secure against relevant security attacks.

A. Authentication Proof based on BAN Logic

In this section, the security of the proposed protocol is analyzed using BAN logic. BAN logic is a well-known security verification and analysis model. It has been widely used for analyzing the security of authentication and session key agreement protocols. Some preliminaries and notations of BAN logic:

a) *Principals* are those agents involved in the protocol (usually people or programs).
b) *Keys* are used to encrypt messages symmetrically.
c) *Public Keys* are similar to keys except that they are used in pairs.
d) *Nonces* are message parts that are not meant to be repeated.
e) *Timestamps* are similar to nonce in that they are unlikely to be repeated.

Relevant BAN logic statements that are useful for analyzing security of the proposed protocol are:

R1: P $|\equiv$ X: P believes X or P would be entitled to believe X. In particular, P can take X as true
R2: P $\lhd$ X: P sees X. P has received some message X and is capable of reading and repeating it.
R3: P $|\sim$X: P once said X. P at some time sent a message including the statement X. It is not known whether this is a replay, though it is known that P believed X when it was sent.
R4: P $\Rightarrow$ X: P has jurisdiction over X. The principal P is an authority on X and should be trusted on this matter.
R5: $\sharp$(X): The message X is fresh.
R6: **(**X, Y): The formulae X or Y is one part of the formulae (X, Y).
R7: $\langle X \rangle_Y$: The formulae X combined with the formulae Y.
R8: $\{X\}_K$: The formulae X is encrypted under the formulae K.
R9: $(X)_K$: The formulae X is hashed with the key K.
R10: P $\xleftrightarrow{K}$ Q: Principal P and Q communicate via shared key K.
R11: P $\Leftrightarrow$ Q: The formulae X is a secret known only to P and Q only and possible to principal trusted by them.

R12: SK: The session key used in the current session.

Relevant logical postulates of BAN logic for this work are:

- The message-meaning rule: $\frac{P \xleftrightarrow{K} Q, P \triangleleft X}{P \mid\equiv Q \mid\sim X}$,

if the principal P believes that the secret key K is shared with the principal Q and P receives the message X encrypted with K then, P believes that the principal Q once sent the message X.

- The freshness-conjuncatenation rule: $\frac{P \mid\equiv \neq (X)}{P \mid\equiv \neq (X,Y)}$,

if the principal believes that X is fresh, then the principal P believes freshness of (X, Y).

- The belief rule: $\frac{P \mid\equiv (X), P \mid\equiv (Y)}{P \mid\equiv (X,Y)}$,

if the principal P believes X and Y, then the principal P believes (X, Y).

- The nonce verification rule: $\frac{P \mid\equiv \neq (X), P \mid\equiv Q \mid\sim X}{P \mid\equiv Q \mid\equiv X}$,

if the principal P believes that X is fresh and the principal Q once sent X then, principal P believes that Q believes X.

- The jurisdiction rule: $\frac{P \mid\equiv Q \Rightarrow X, P \mid\equiv Q \mid\equiv X}{P \mid\equiv X}$,

if the principal believes that Q has jurisdiction over X and Q believes X, then P believes that X is true.

- The session key rule: $\frac{P \mid\equiv \neq (X), P \mid\equiv Q \mid\equiv (X)}{P \mid\equiv P \xleftrightarrow{K} Q}$,

if the principal P believes that the session key is fresh and the principal P and Q believes X, which are the necessary parameters of the session key, then principal P believes that he/she shares the session key K with Q.

In order to prove the proposed protocol secure, the proposed protocol must satisfy the following goals based on BAN logic, where RS and U_i define registration server and consumer respectively.

Goal 1: $U_i \mid\equiv U_i \xleftrightarrow{SK} RS$

Goal 2: $U_i \mid\equiv RS \mid\equiv U_i \xleftrightarrow{SK} RS$

Goal 3: $RS \mid\equiv RS \xleftrightarrow{SK} U_i$

Goal 4: $RS \mid\equiv U_i \mid\equiv RS \xleftrightarrow{SK} U_i$

The proposed protocol is transformed to the idealized form as:

$MSG_1 : U_i \rightarrow RS : \langle ID_i, M_5, T_i \rangle : \langle M_1 \rangle_{G_i}$

$MSG_2 : RS \rightarrow U_i : \langle M_7 \rangle : \langle r_j \rangle_{K_x}$

The following assumptions about the initial state of the protocol are given:

$ASM_1 : U_i \mid\equiv \neq (r_i, r_j)$

$ASM_2 : RS \mid\equiv \neq (r_j, r_i)$

$ASM_3 : U_i \mid\equiv U_i \xleftrightarrow{G_i} RS$

$ASM_4 : RS \mid\equiv RS \xleftrightarrow{Kx} U_i$

$ASM_5 : U_i \mid\equiv RS \Rightarrow r_j$

$ASM_6 : RS \mid\equiv U_i \Rightarrow r_i$

Applying BAN logic rules and assumptions:
$MSG_1 : U_i \rightarrow RS : \langle ID_i, M_5, T_i \rangle : \langle M_1 \rangle_{G_i}$

Thus
S1: $RS \lhd \langle ID_i, M_5, T_i \rangle : \langle M_1 \rangle_{G_i}$

Applying assumption ASM_4, S1 and message meaning rule gives:
S2: $RS \mid\equiv U_i \sim M_1$

According to ASM_2, S2, freshness-conjuncatenation and nonce verification rule:
S3: $RS \mid\equiv U_i \mid\equiv M_1$, where information of the parameter M_1 is used to computed the session key in our protocol.

According to ASM_6, S3 and jurisdiction rule:
S4: $RS \mid\equiv M_1$

According to ASM_2, S3 and session key rule:
S5: $RS \mid\equiv RS \xrightarrow{SK} U_i$ (***Goal 3)***

According to ASM_2, S5 and nonce verification rule:
S6: $RS \mid\equiv U_i \mid\equiv RS \xrightarrow{SK} U_i$ (***Goal 4)***

$MSG_2 : RS \rightarrow U_i : \langle M_7 \rangle : \langle r_j \rangle_{K_x}$
According to seeing rule:
S7: $U_i \lhd : \langle M_7 \rangle : \langle r_j \rangle_{K_x}$
Applying the assumption ASM_3, S7 and message meaning rule:
S8: $U_i \mid\equiv RS \sim r_j$
According to ASM_1, S8, freshness-conjuncatenation and nonce verification rule:
S9: $U_i \equiv RS \mid\equiv r_j$, where information of the parameter r_j is used to computed session key in our protocol.

According to ASM_5, S9 and jurisdiction rule:
S10: $U_i \mid\equiv r_j$

According to ASM_1, S9 and session key rule:
S11: $U_i \mid\equiv U_i \xrightarrow{SK} RS$ (***Goal 1)***

According to ASM_1, S11 and nonce verification rule:
S12: $U_i \mid\equiv RS \mid\equiv U_i \xrightarrow{SK} RS$ (***Goal 2)***

The above justification claims that the declared goals have been successfully proven using BAN logic model. Therefore, it can be claimed that the proposed protocol successfully provides mutual authentication property as well as session key negotiation between the user and RS.

B. Further Security Analysis

It has been observed that numerous authentication protocols [1], [2], [13], [14], [17], [20] analyze the resilience against known attacks through informal security analysis [21], [22]. Therefore, this section provides the description of the resilience against the known security attacks, such as off-line password guessing attack, privileged insider attack, user impersonation attack, server impersonation attack, known key security attack, stolen-verifier attack, DoS attack and mutual authentication.

1) Off-line password guessing attack

During the registration phase, U_i's password PW_i was never transmitted to RS in plaintext form and the computation of PWB_i depends on PW_i and random number b_i. Therefore, if the adversary wants to guess PW_i, they have to first know PWB_i, which is used to compute M_5 in *Step 2* of mutual authentication and session key negotiation phase, where $M_5 = E_{K_x}\left(M_1 \| M_4 \| PWB_i \| C_i\right)$ and PWB_i is encrypted with key K_x. Thus, the adversary cannot retrieve PWB_i without K_x. Accordingly, the adversary cannot compute PWB_i using M_6 without K_y, where $M_6 = h\left(ID_i \| PWB_i \| K_x^{'} \| r_j^{'} \| T_j\right)$. Hence, this proposed protocol claims that it is immune to the password guessing attack.

2) Privileged insider attack

During the registration, as mentioned in the literature [5], [6], a user's password should not be sent to RS in plaintext form during the registration phase in order to resist the insider attack. In the registration phase of this work, U_i sends a masked password PWB_i to RS instead of PW_i, where $PWB_i = h\left(PW_i \| b_i\right)$. Therefore, the insider attach of RS cannot extract PW_i from PWB_i due to the strong collision resistance property of the hash function $h()$.

3) User impersonation attack

Suppose that an adversary endeavors to impersonate U_i. In order to do it, the adversary first captures U_i's message from the public channel and then makes an effort to generate another valid message, which should be authenticated by RS. The adversary traps $\langle ID_i, M_5, T_i \rangle$ from the public channel and tries to compute $\langle M_2, K_y, C_i \rangle$ using the known information. However, the adversary cannot compute M_2 and K_y without θ_i and x, respectively, where x is the secret key of RS. In addition, C_i is also secure being stored in the registration software in U_i in encrypted form. Therefore, it is difficult task for the adversary to impersonate U_i.

4) Server impersonation attack

An adversary may try to impersonate RS in the mutual authentication phase. In this proposes protocol, RS sends $\langle M_7 \rangle$ to the registration software in U_i through an open channel, where $M_7 = E_{K_y^{'}}\left(M_6 \| r_j\right)$. Note that $\langle M_7 \rangle$ is encrypted with key K_y and it is depends on M_6 and r_j, where $M_6 = h\left(ID_i \| PWB_i \| K_x^{'} \| r_j \| T_j\right)$. It is clear that the adversary can easily generate a random number, but to compute M_6, the adversary needs (PWB_i, K_x). However, the adversary is unable to successfully compute $\left(PWB_i, K_x\right)$ from the public message. Therefore, this proposed protocol can withstand the server impersonation attack.

5) Stolen-verifier attack

This type of attack occurs when the stored information in RS is leaked, however, the authentication system should not be affected by the adversary. Suppose that the information stored in the table available to RS has been compromised, where the table contains the entries of the form $\left(ID_i, UNSID_i, SL_i, \left(SK_i \oplus h\left(ID_i \oplus x\right)\right)\right)$. Note that the adversary cannot extract $h\left(ID_i \oplus x\right)$ without SK_i. Furthermore, a valid user is not able to obtain long-term information from RS. Therefore, the adversary is unable to get any advantage after obtaining the stored table.

6) Denial-of-service attack

In biometric based authentication, the biometric information may be affected due to noise during the biometric acquisition, resulting in difficulty in reproducing the exact biometric data signature accurately each time. The hash function is very sensitive to even slight changes in the input. Therefore, the hash function cannot be applied directly to the biometric data. A legal user may even fail to login to the remote server due to noisy biometric sensor data. If a biometric based authentication protocol relies on verifying $h\left(BT_i^*\right) = ?\, hBT_i$, in each session, then U_i may get rejected and in biometric authentication this phenomenon is called the DoS attack. In order to resist such kind of problem, a fuzzy extractor is typically used. Therefore, the registration software in U_i passes the biometric verification of U_i and thus, it can withstand the DoS attack.

7) Mutual authentication

Mutual authentication [23] is typically one of the important and enviable property of any client-server authentication protocol. In *Step 3* of the mutual authentication phase of this work, RS verifies the authenticity of U_i by checking the condition $M_3^{'} = ?\, M_3$ whereas U_i checks $M_6^{'} = ?\, M_6$ in *Step 5* to verify the legitimacy of RS. Therefore, this proposed protocol achieves the mutual authentication property.

8) Man-in-the-middle attack

In this form of attack, the adversary ensnares the public messages and attempts to act as a middle broker between the user and the remote server. In user impersonation attack, the work demonstrated that the adversary cannot generate a forged login message without knowing the user's secret information. For the same reason, the adversary cannot also impersonate the RS. Therefore, this proposed protocol can withstand the man-in-the-middle attack.

VI. PERFORMANCE ANALYSIS

This section appraises the performance of the proposed protocol in terms of computation and communication costs with other competitive protocols [7], [9], [10]. This work uses crypto-operations to evaluate the computation cost. The notations and description of the crypto-operations are:

- T_e: Time needed to perform exponentiation operation.
- T_{pm}: Time needed to perform elliptic curve point multiplication operation.
- T_h: Time needed to perform one-way hash operation.
- T_s: Time needed to perform symmetric key encryption/decryption operation.

TABLE II provides computation costs of this proposed protocol compared to the relevant literature [7], [9], [10]. This proposed protocol requires an increased computation cost, however for the considered device, the increase in computation cost is marginal compared to the significantly improved security benefits.

The communication cost of this work compared to the literature [7], [9], [10] was analyzed. It was observed that this proposed protocol has a lower communication cost than the protocols considered in the literature. For comparison purposes, this work assumed that the length of ID_i, PW_i and BT_i are 64 bits of length each. In addition, the message digest of the hash function, ECC-point multiplication and symmetric key encryption produced 160-bits, 160-bits and 128-bits, respectively. TABLE III presents the communication overhead cost and it can be observed that the proposed protocol is very efficient in terms of the communication cost.

TABLE II
COMPARISON OF THE COMPUTATIONAL COST OF THIS WORK COMPARED TO THE LITERATURE

	User cost	Server cost	Total cost
Yang *et al*. [7]	$4T_e + 3T_h + 1T_s$	$6T_e + 2T_h + 1T_s$	$10T_e + 5T_h + 2T_s$
Lee *et al*. [9]	$2T_{pm} + 5T_h + 1T_s$	$2T_{pm} + 4T_h + 1T_s$	$4T_{pm} + 9T_h + 2T_s$
He *et al*. [10]	$2T_{pm} + 5T_h + 1T_s$	$2T_{pm} + 4T_h + 1T_s$	$2T_{pm} + 4T_h + 1T_s$
Proposed	$3T_{pm} + 5T_h + 3T_s$	$3T_{pm} + 5T_h + 2T_s$	$6T_{pm} + 10T_h + 5T_s$

TABLE III
COMPARISON OF THE COMMUNICATION COST OF THIS WORK COMPARED TO THE LITERATURE

	User	Server	Total cost
Yang *et al*. [7]	4224	1312	5536
Lee *et al*. [9]	480	480	960
He *et al*. [10]	480	480	960
Proposed	256	256	512

VII. CONCLUSION

The main intention of this paper is to provide security protection on the stored information in the consumer device from the unauthorized access by implementing an authentication protocol. In order to do it, this paper proposes a mutual authentication and key negotiation protocol using elliptic curve cryptography. The security verification of the protocol has been done using BAN logic and the security analysis ensures that the protocol can withstand several relevant security attacks. The protocol is not only efficient in terms of security attacks, but it also achieves high performance in terms of communication cost in comparison with the existing protocols. Moreover, the proposed protocol provides the mutual authentication property between the participants involved and provides a password update facility to registered users. This work enables secure biometric personal storage devices to be configured from an Internet service and maintained throughout the lifetime of the device.

REFERENCES

[1] L. Lamport, "Password authentication with insecure communication," *Communications of the ACM*, vol. 24, no. 11, pp. 770–772, Nov. 1981.

[2] M.-S. Hwang, and L.-H. Li, "A new remote user authentication scheme using smart cards," *IEEE Trans. Consumer Electron.*, vol. CE-46, no. 1, pp. 28–30, Feb. 2000.

[3] H.-M. Sun, "An efficient remote use authentication scheme using smart cards," *IEEE Trans. Consumer Electron.*, vol. CE-46, no. 4, pp. 958–961, Nov. 2000.

[4] C.-K. Chan, and L.M. Cheng, "Cryptanalysis of a remote user authentication scheme using smart cards," *IEEE Trans. Consumer Electron.*, vol. CE-46, no. 4, pp. 992–993, Nov. 2000.

[5] R. Amin, and G. P. Biswas, "A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks," *Ad Hoc Networks*, vol. 36, no. 1, pp. 58–80, Jan. 2016.

[6] R. Amin, and G. P. Biswas, "A novel user authentication and key agreement protocol for accessing multi-medical server usable in TMIS," *Journal of Medical Systems*, vol. 39, no. 3, pp. 1–17, Mar. 2015.

[7] F.-Y. Yang, T.-D. Wu, and S.-H. Chiu, "A secure control protocol for USB mass storage devices," *IEEE Trans. Consumer Electron.*, vol. CE-56, no. 4, pp. 2339–2343, Nov. 2010.

[8] B. Chen, C. Qin, and L. Yu, "A Secure Access Authentication Scheme for Removable Storage Media," *Journal of Information & Computational Science*, vol. 9, no. 15, pp. 4353–4363, Nov. 2012.

[9] C. Lee, C. Chen, and P. Wu, "Three-factor control protocol based on elliptic curve cryptosystem for universal serial bus mass storage devices," *IET Computers & Digital Techniques*, vol. 7, no. 1, pp. 48–55, Jan. 2013.

[10] D. He, N. Kumar, J.-H. Lee, and R. S. Sherratt, "Enhanced three-factor security protocol for consumer USB mass storage devices," *IEEE Trans. Consumer Electron.*, vol. CE-60, no. 1, pp. 30–37, Feb. 2014.

[11] D. Giri, R. S. Sherratt, T. Maitra, and R. Amin, "Efficient Biometric and Password Based Mutual Authentication for Consumer USB Mass Storage Devices," *IEEE Trans. Consumer Electron.*, vol. CE-61, no. 4, pp. 491–499, Nov. 2015.
[12] D. Giri, R. S. Sherratt, and T. Maitra, "A novel and efficient session spanning biometric and password based three-factor authentication protocol for consumer USB mass storage devices," *IEEE Trans. Consumer Electron.*, vol. CE-62, no. 3, pp. 283–291, Aug. 2016.
[13] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *LNCS*, vol. 3027, pp. 523–540, 2004.
[14] X. Boyen, "Reusable cryptographic fuzzy extractors," in *Proc. ACM CCS*, 2004, pp. 82–91.
[15] R. Amin, and G.P. Biswas, "Anonymity preserving secure hash function based authentication scheme for consumer USB mass storage device," in *Proc. IEEE CCCIT*, 2015, pp. 1–6.
[16] N. Koblitz, "Elliptic curve cryptosystem," *Mathematics of Computation.*, vol. 48, no. 177, pp. 203–209, Jan. 1987.
[17] V. S. Miller, "Use of elliptic curves in cryptography," *LNCS*, vol. 218, pp. 417–426, Dec. 2000.
[18] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *ACM Trans. Computer Systems*, vol. 8, no. 1, pp. 18–36, Feb. 1990.
[19] D. He, N. Kumar and N. Chilamkurti, "A secure temporal-credential-based mutual authentication and key agreement scheme with pseudo identity for wireless sensor networks," *Information Sciences*, vol. 321, pp. 263–277, Nov. 2015.
[20] P. Sarkar, "A simple and generic construction of authenticated encryption with associated data," *ACM Trans. Information and System Security*, vol. 13, no. 4, pp. 1–16, Dec. 2010.
[21] W.-C. Ku, and S.-M. Chen, "Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards," *IEEE Trans. Consumer Electron.*, vol. CE-50, no. 1, pp. 204–207, Feb. 2004.
[22] E.-J. Yoon, E.-K. Ryu, and K.-Y. Yoo, "Further improvement of an efficient password based remote user authentication scheme using smart cards," *IEEE Trans. Consumer Electron.*, vol. CE-50, no. 2, pp. 612–614, May. 2004.
[23] S. H. Islam, and G. P. Biswas, "A more efficient and secure ID-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem," *J. Systems and Software*, vol. 84, no. 11, pp. 1892–1898, Nov. 2011.

Ruhul Amin received his B.Tech and M.Tech from West Bengal University of Computer Science and Engineering, Indian Engineering in 2009 and 2013, respectively. He was a Ph.D. research scholar in Computer Science and Engineering, Indian School of Mines (ISM), Dhanbad, India. He is currently a Lecturer in the Department of Computer Science and Engineering, Thapar University, Patiala, Punjab, India. He has published many research papers in Journals and Conference proceedings of International reputes. His current research interests include cryptographic authentication protocols and security in wireless sensor networks.

R. Simon Sherratt (M'97-SM'02-F'12) received the B.Eng. degree in Electronic Systems and Control Engineering from Sheffield City Polytechnic, UK in 1992, M.Sc. in Data Telecommunications in 1994 and Ph.D. in video signal processing in 1996 from the University of Salford, UK.

In 1996, he has appointed as a Lecturer in Electronic Engineering at the University of Reading where he is now Professor of Biosensors. His research topic is signal processing and personal communications in consumer devices focusing on wearable devices and healthcare.

He received the 1st place IEEE Chester Sall Memorial Award in 2006, the 2nd place in 2016 and the 3rd place in 2017. He is a reviewer for the IEEE SENSORS JOURNAL and is currently a Senior Editor and Emititus Editor-in-Chief of the IEEE TRANSACTIONS ON CONSUMER ELECTRONICS.

Debasis Giri received the Ph.D degree from the Indian Institute of Technology, Kharagpur, India in 2009. He did his masters (M.Tech and M.Sc) both from Indian Institute of Technology, Kharagpur, India in 2001 and 1998 respectively. Presently he is a Dean under the school of Electronic, Computer Science and Informatics of Haldia Institute of Technology, India, and Professor in the Department of Computer Science and Engineering, Haldia Institute of Technology, India. He has tenth All India Rank with percentile score 98.42 in the Graduate Aptitude Test in Engineering (GATE) Examination in 1999. His current research interests include Cryptography, Network security, Data Hiding, Security in Wireless Sensor Networks and Security in VANETs.

Dr. Giri is an Editorial Board Member and a Reviewer of many reputed International Journals. Presently he is an Associate Editor of the Journal of Security and Communication Networks (Wiley), the Journal of Communication Systems (Wiley) and the Journal of Electrical and Computer Engineering Innovations. He is also a Program Committee member for many International Conferences.

SK Hafizul Islam received the M.Tech from ISM Dhanbad in 2009 and the Ph.D in Computer Science and Engineering from Indian School of Mines, Dhanbad (SM Dhanbad), India. He was an Assistant Professor in the Department of CSIS, BITS Pilani, Pilani Campus, Rajasthan, India and is currently an Assistant Professor in the Department of CSE, Indian Institute of Information Technology, Kalyani (IIIT Kalyani), West Bengal, India. He has published 50 research papers in reputed international Journals and Conference proceedings. He is an Associate Editor of the International journal of Communication Systems, Wiley. His research interest includes Cryptography and Information Security.

Muhammad Khurram Khan (M'07, SM'12) is currently working as a Full Professor at the Center of Excellence in Information Assurance (CoEIA), King Saud University, Kingdom of Saudi Arabia. He has published over 250 research papers in the journals and conferences of international repute. In addition, he is an inventor of 10 US/PCT patents.

Prof. Khan is the Editor-in-Chief of Telecommunication Systems Journal, Springer. He is a Fellow of the IET, Fellow of the BCS, Fellow of the FTRA, a member of the IEEE Technical Committee on Security & Privacy, a member of the IEEE Cybersecurity community, and a member of IEEE Consumer Electronics society.