A Lightweight Pseudonym Authentication and Key Agreement Protocol for Multi-medical Server Architecture in TMIS

Xiaoxue Liu[1], Yanping Li[1], Juan Qu[2], and Yong Ding[3]
[1]School of Maths. and Info. Science, Shaanxi Normal University
Xi'an, China, 710119
[e-mail: 862417756@qq.com,lyp@snnu.edu.cn]
[2]School of Maths. and Stats., Chongqing Three Gorges University
Chongqing,China, 404100
[e-mail: qulujuan@163.com]
[3]School of Computer Sci. and info. security, Guangxi Key Laboratory of Cryptography and Info. Security,
Guilin,China, 541004
[e-mail: stonedingy@126.com]
*Corresponding author: Yanping Li

Received September 19, 2016; revised November 17, 2016; accepted December 13, 2016; published February 28, 2017

Abstract

Telecare Medical Information System (TMIS) helps the patients to gain the health monitoring information at home and access medical services over the mobile Internet. In 2015, Das et al proposed a secure and robust user AKA scheme for hierarchical multi-medical server environment in TMIS, referred to as DAKA protocol, and claimed that their protocol is against all possible attacks. In this paper, we first analyze and show DAKA protocol is vulnerable to internal attacks, impersonation attacks and stolen smart card attack. Furthermore, DAKA protocol also cannot provide confidentiality. We then propose a lightweight pseudonym AKA protocol for multi-medical server architecture in TMIS (short for PAKA). Our PAKA protocol not only keeps good security features declared by DAKA protocol, but also truly provides patient's anonymity by using pseudonym to protect sensitive information from illegal interception. Besides, our PAKA protocol can realize authentication and key agreement with energy-saving, extremely low computation cost, communication cost and fewer storage resources in smart card, medical servers and physical servers. What's more, the PAKA protocol is proved secure against known possible attacks by using Burrows-Abadi-Needham (BAN) logic. As a result, these features make PAKA protocol is very suitable for computation-limited mobile device.

Keywords: multi-medical server; privacy-preserving; authentication; BAN logic

This work was supported by the National Natural Science Foundation of China under Grant 61402275,61402015,6157224, 61272436,Shaanxi Province Natural Science Basic Research Program 2016JM6069, the Scientific Research Foundation for the Returned Overseas Chinese Scholars of MOHRSS, the Fundamental Research Funds for the Central Universities under Grant GK201603012,GK201402004, the Innovation Fund Designated for Graduate Students of Shaanxi Normal University(2015CXS022).

https://doi.org/10.3837/tiis.2017.02.016 **ISSN : 1976-7277**

1. Introduction

Telecare Medical Information System (TMIS) builds a convenient connection between patients and doctors and helps the patients to gain the health monitoring information at home and access medical services over the mobile Internet. With the increasing dependence on the Internet, however, the single-medical server cannot meet people's basic needs. The research [1] found that one user averagely logins into 25 different servers in each month. The traditional authentication protocol is in a single medical server environment where each patient has to repetitively register in the different servers and remember numerous different usernames and passwords for different medical servers. It is especially inconvenient and inefficient for the patients in TMIS. In the real TMIS, many patients may use the same username and password to access different medical servers simply for convenience, this easily increases the risk of disclosure of patient's usernames and passwords. Once the patient's usernames and passwords are leaked out and occasionally got by an attacker, the attacker may use the compromised username and password to login to other medical servers the patient has registered. Thus, if there is a system where a user can login to several medical servers to access different medical services only by one username and a password, it is convenient for the patients. Luckily, multi-medical server system can realize the above hypothesis and solve the problem of repeated registration which is inherent in single-medical server scenarios of TMIS. Meanwhile, TMIS can provide various resources to the patient like health educators, physicians, care-givers, public health organizations and home-care service. TMIS can get a mass of user data from different servers, which increases the leakage likelihood of user privacy. Obviously, patients' privacy protection is an urgent demanding in medical environment. The protocols designed for TMIS should take users' privacy-protection into account. At same time, most patients are mobile phone users. So computation-limited and energy-limited problems are also inevitable. Therefore, it is very important and urgent to design patients' securely and efficiently remote authentication protocols in multi-medical server environment for TMIS.

1.1 Related Work

Due to the widespread applications of Internet multi-medical servers and the great convenien -ce of remote medical services, how to securely access the remote medical servers and get the corresponding service has received considerable attention. In recent years, many remote AKA protocols are successively proposed in Telecare Medical Information System (TMIS) [2-9]. Wu et al [2] proposed a novel authentication protocol for TMIS. However, He et al [3] shows that Wu et al's protocol [2] cannot resist insider attacks and impersonation attacks. In 2012, Wei et al. [4] showed that both of protocols in [2] and [3] failed to meet multi-factor authentication and further proposed an improved protocol at same time. Thereafter, Zhu et al [5] described Wei et al's protocol [4] is vulnerable to off-line password guessing attack. Then, Lee-Liu [6] demonstrated that the new protocol in [5] cannot withstand parallel session attack and presented an improved one. In 2013, Tan et al. [7] proposed an efficient biometrics-based authentication scheme for TMIS, which was claimed to resist many kinds of attacks. However, Yan et al. [8] declared that the protocol in [7] is vulnerable to Dos attack. In order to eliminate the drawbacks in [7], a new scheme [8] proposed for better security protection and performance. Unfortunately, Mishra et al. [9] shows that Yan et al's scheme [8] suffers from password guessing attack and they also proposed a securely enhanced protocol. However, all schemes above are suitable for single-medical server environment. They cannot meet the

various requirements of people and the rapid development of multi-medical servers. In last few years, a large number of user authentication protocols for multi-server system have been proposed [10-14]. Thought the protocols in [10-14] have some advantages (such as strong-anonymity), these authentication protocols need heavy calculations because of the public encryption/signature algorithms or other time-consuming computation (such as bilinear pairing). Therefore, these protocols are not suitable for the energy-limited mobile devices. Consider a huge number of mobile terminal users have limit computation and energy (battery-powered), they frequently login in the remote medical servers according to their needs. The lightweight remote AKA protocols are urgently required. Because hashing operations and XOR operations require very little computations and energy, the lightweight remote AKA protocols only by using hashing operations and XOR operations are significant. In other words, the efficient and energy-saving AKA protocols keep pace with the development of the mobile Internet. Amin et al. [15] first proposed a novel AKA protocol for accessing remote multi-medical server in TMIS, which was claimed to resist many kinds of attacks. However, Das et al. proposed a new protocol (abbreviated DAKA) [16] and showed Amin et al's scheme [15] is vulnerable to internal attack, replay attack and the man-in-middle attack. The DAKA protocol further proposed an improved protocol in order to overcome the flaws in Amin et al's protocol, and claimed that their protocol is against all possible attacks. Unfortunately, after careful analysis, we found the DAKA [16] protocol still suffers from internal attack, impersonation attack and stolen smart card attack. Furthermore, it also cannot provide confidentiality. In order to fix the flaws, a lightweight pseudonym authentication and key agreement (PAKA, for short) protocol for multi-medical server architecture in TMIS is proposed in this paper.

1.2 Our Contributions

In our PAKA, the patient U_i can remotely log in the physical server PS_{jk} who is under the jurisdiction of the medical server MS_j. U_i and MS_j need to register at the registration center MRS in advance. And in AKA progress, MS_j authenticates U_i, and sends U_i's login request to PS_{jk}. After verifying the validity of MS_j's message, PS_{jk} directly sends message to U_i. Hereafter, U_i and PS_{jk} not only realize mutual authentication but also establish a session key. Compared with [16], the PAKA protocol not only needs lower computational consumption and communication consumption, but also can provide the following security features.

- First, the PAKA protocol can provide user's anonymity to protect patient's privacy by randomized pseudonym. The medical server and physical server only verify that the authenticated patient is a legal patient, but do not know his true identity. Hence, our PAKA protocol is practical in the privacy enhanced scenarios.
- Second, the PAKA protocol can realize authentication and key agreement among the mobile terminal patients, different remote medical servers and physical servers only by using hashing and XOR operations, both of which require little computation and energy cost, storage overhead for mobile terminals patients. A patient can login in several different medical servers to obtain different medical services by using only one single username and a password without repeated registration problem. The whole protocol still adopts the classic three handshakes of 'request-challenge-response' and does not increase interaction numbers and communication overload. Compared with other exiting protocols in TMIS (showed in Table 2), the PAKA protocol is more lightweight and efficient. Hence, it is very suitable for computation-limited mobile devices.

- Third, the PAKA protocol can provide three-factor authentication including the smart card (something the user has), password (something the user knows) and biometric key (something the user is). Because biometric key is difficult to lose, forget, copy, share, guess or break, it is believed to be a reliable authentication factor [17], [18]. In our PAKA protocol, the smart card is used to authenticate the cardholder. Only the entered identity, password and biometric key all are correct, then the smart card can be activated and interact with the remote medical servers to help patient with authentication. The biometric key is obtained by a fuzzy extractor which can output the same random string when the input is close to the original biometric information. It make our protocol is more robust and fault-tolerant.
- Fourth, by using the BAN logic, the PAKA protocol is proved secure against possible known attacks and satisfies the secure requirements of AKA protocols for multi-medical server architecture. Hence, the PAKA is practical in complex network environment.

The rest of paper is organized as follows. Some mathematical preliminaries about fuzzy biometrics extractor is introduced in Section 2. Section 3 briefly reviews the DAKA protocol and Section 4 analyses its weaknesses. The PAKA protocol is presented in Section 5. Detailed security analysis and proof are given in Section 6. The comparisons of the performance and security features between PAKA protocol with other related schemes are discussed in Section 7. Section 8 concludes this paper.

2. Preliminaries

Here, we briefly introduce the mathematical preliminaries about biometrics and fuzzy extractor [10], [11], [19]. The fuzzy extractor is a tuple (M, l, t) consisted by two procedures: the probabilistic generation procedure (Gen) and the deterministic reproduction procedure (Rep).

- Gen is a probabilistic generation procedure, which on input biometric data $B_i \in M$, outputs an extracted string $\sigma_i \in \{0,1\}^l$,and an auxiliary string $\tau_i \in \{0,1\}^l$, where $l = |\sigma_i|$, $(\sigma_i, \tau_i) = Gen(B_i)$.
- Rep is a deterministic reproduction procedure that allows to recover σ_i from the corresponding auxiliary string τ_i,and any vector B_i' close to B_i, where $\sigma_i = Rep(B_i', \tau_i)$, for all B_i, B_i' satisfying $dis(B_i, B_i') \le t$ (t is the tolerance threshold).

The uniqueness of individual biological information makes it suitable for authentication protocols. Compared with poor password, biometric key has more advantages [20],[21]. Thus, the probability to guess the biometric data σ_i by an attacker is approximately $\frac{1}{2^l}$ [11].

3. Review of the DAKA Protocol

The DAKA protocol is composed of **Registration, Login, Authentication, Password** and **Biometric Update and Dynamic Medical Server Addition** [16], which is shown in **Fig. 1**. To simplify the subsequent description, some notations are given in **Table 1**. At the beginning, the medical registration server MRS selects his private key X_r, where $|X_r| = l$ and a

cryptographically secure one-way hash function $h(\cdot): \{0,1\}^* \rightarrow \{0,1\}^l$. Medical server MS_j sets up the secret session key X_{jk} with each physician server PS_{jk} respectively, where $|X_{jk}|=l$. The DAKA protocol is briefly reviewed as follows.

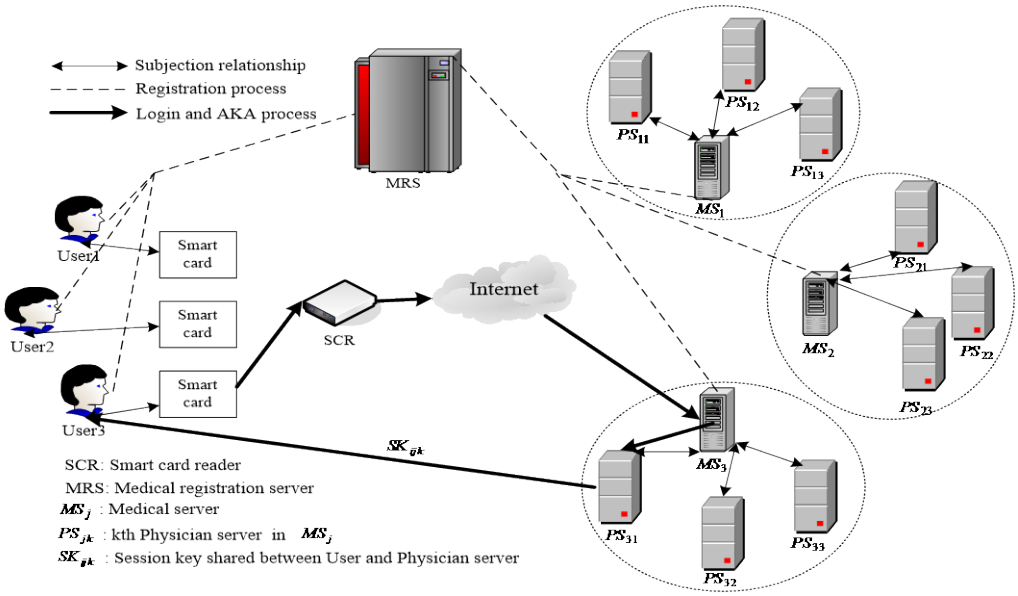

Fig. 1. Architecture for accessing multi medical servers in TMIS

3.1 Registration phase

1) **Medical Server Registration Phase:** If a medical server MS_j wants to provide the medical services to the remote patients, he/she needs to register in MRS firstly.

Rs1 MS_j chooses his/her ID_{S_j}. Then, $MS_j \Rightarrow MRS: ID_{S_j}$;

Rs2 Upon receiving the registration message ID_{S_j}, MRS computes $X_j = h(ID_{S_j} \| X_r)$. Then, $MRS \Rightarrow MS_j: X_j$;

Rs3 After receiving the message X_j, MS_j keeps (ID_{S_j}, X_j) secretly.

2) **User Registration Phase:** A patient U_i needs to register in MRS with the following steps if he wants to get remote medical services:

Ru1 U_i chooses his/her ID_i, PW_i and a random number R_i, gets personal biometric data B_i, and computes biometric key σ_i by $(\sigma_i, \tau_i) = Gen(B_i)$, $RPW_i = h(ID_i \| R_i \| PW_i)$. Then, $U_i \Rightarrow MRS: ID_i, RPW_i$;

Ru2 On receiving the registration message from U_i, MRS computes $A_j = h(ID_i \| X_j) \oplus RPW_i$, $P_j = h(ID_{S_j} \| X_j) \oplus RPW_i$, where $1 \le j \le m + m'$, where m' is reserved space to increase the number of servers in the future. Stores $\{\{ID_{S_j}, A_j, P_j\}, Gen(\cdot), Rep(\cdot), h(\cdot), t\}$ in smart card, where t is the error −tolerance threshold of fuzzy extractor. Then, $MRS \Rightarrow U_i$: Smart card;

Ru3 U_i computes $e_i = h(ID_i \| \sigma_i) \oplus R_i$, $f_i = h(ID_i \| RPW_i \| \sigma_i)$. Finally $\{\{ID_{S_j}, A_j, P_j\}, e_i, f_i, \tau_i, Gen(\cdot), Rep(\cdot), h(\cdot), t\}$ are stored in U_i's smart card

Table 1. Notations

Symbol	Description
U_i	the i th patient(user) who can access medical services from the physician servers with the help of MS_j
MRS	medical registration server which is responsible for providing registration to new users/patients as well as MS_j, $j = 1,2,.....,m$.
MS_j	the j th medical server
PS_{jk}	k th physician server in MS_j
ID_i, ID_{S_j},	identity of U_i, MS_j PS_{jk}
X_r	master secret key hold by MRS
X_{jk}	secret session key between MS_j and PS_{jk}
z_j	master secret key hold by MS_j
$h(\cdot)$	a cryptographically secure one way hash function
$\oplus, \|\|$	bitwise XOR operation and concatenation operation
$\rightarrow$	a public communication channel
$\Rightarrow$	a secure communication channel

3.2 Login phase

L1 U_i inserts his/her smart card into the card reader, and inputs his/her ID_i, PW_i, $B_i^{'}$;

L2 The smart card computes $\sigma_i = Rep(B_i^{'}, \tau_i)$, $R_i = e_i \oplus h(ID_i \| \sigma_i)$, $RPW_i = h(ID_i \| R_i \| PW_i)$, and checks $f_i ? = h(ID_i \| RPW_i \| \sigma_i)$. If it does not match, the session is terminated. Otherwise;

L3 The smart card generates a random number R_c, the current time-stamp TS_c, and computes $M_1 = A_j \oplus RPW_i = h(ID_i \| X_j)$, $M_2 = P_j \oplus RPW_i = h(ID_{S_j} \| X_j)$, $M_3 = ID_i \oplus M_2$, $M_4 = ID_i \oplus M_1 \oplus R_c$, $M_5 = h(M_1 \| M_3 \| M_4 \| R_c \| TS_c)$. Then, $U_i \rightarrow MS_j$: **msg1=** $\{ID_{S_j}, ID_{jk}, M_3, M_4, M_5, TS_c\}$;

3.3 Authentication phase

V1 Upon receiving **msg1**, MS_j reads the current time-stamp TS_c^*, and checks $|TS_c^* - TS_c| \leq \Delta T$. If the verification fails, the request is rejected. Otherwise, MS_j computes $M_6 = h(ID_{S_j} \| X_j) = M_2$, $M_7 = M_3 \oplus M_6 = ID_i$, $M_8 = h(M_7 \| X_j)$, $M_9 = M_4 \oplus M_7 \oplus M_8 = R_c$, and checks $M_{10} = h(M_8 \| M_3 \| M_4 \| M_9 \| TS_c) ? = M_5$ .If it does not hold, MS_j terminates it. Otherwise,

V2 MS_j generates a random number R_s, reads times-tamp TS_s, and computes $M_{11}=h(ID_{S_j} \| ID_{jk} \| X_{jk})$, $M_{12}=ID_i \oplus M_{11}$, $M_{13}=h(ID_i \| X_{jk}) \oplus R_s$, $M_{14}=ID_i \oplus R_c \oplus R_s$, $M_{15}=h(ID_i \| M_{11} \| M_{12} \| M_{13} \| M_{14} \| R_c \| R_s \| TS_s)$. Then, $MS_j \rightarrow PS_{jk}$: **msg2=** $\{ID_{S_j}, ID_{jk}, M_{12}, M_{13}, M_{14}, M_{15}, TS_s\}$**;**

V3 Upon receiving **msg2**, PS_{jk} checks the validity of TS_s, by $|TS_s^* - TS_s| \leq \Delta T$, where TS_s^* is the current time-stamp of PS_{jk}, if it does not hold, PS_{jk} rejects the session. Otherwise, PS_{jk} computes $M_{16}=h(ID_{S_j} \| ID_{jk} \| X_{jk})$, $M_{17}=M_{12} \oplus M_{16}=ID_i$, $M_{18}=M_{13} \oplus h(M_{17} \| X_{jk})=R_s$, $M_{19}=M_{14} \oplus M_{17} \oplus M_{18}=R_c$, and check $M_{20}=h(M_{17} \| M_{16} \| M_{12} \| M_{13} \| M_{14} \| M_{19} \| M_{18} \| TS_s) ?= M_{15}$. If it does not hold, PS_{jk} rejects the session. Otherwise,

V4 PS_{jk} generates a random number R_k, reads timestamp TS_k and computes $M_{21}=h(M_{17} \| X_{jk})$, $M_{22}=ID_i \oplus R_c \oplus R_k$, $M_{23}=M_1 \oplus R_k$, $SK_{ijk}=h(ID_i \| ID_{jk} \| R_c \| R_k \| M_1 \| TS_k)$, $M_{24}=h(SK_{ijk} \| M_{22} \| M_{23} \| R_c \| R_k \| TS_k)$. Then, $PS_{jk} \rightarrow U_i$:**msg3=**$\{ID_{jk}, M_{22}, M_{23}, M_{24}, TS_k\}$**;**

V5 Upon receiving **msg3**, U_i checks the validity of TS_k, by $|TS_k^* - TS_k| \leq \Delta T$, where TS_k^* is the current time-stamp of U_i, if it does not hold, U_i rejects the session, Otherwise, U_i computes $M_{25}=M_{22} \oplus ID_i \oplus R_c=R_k$, $M_{26}=M_{23} \oplus M_{25}=h(ID_i \| X_{jk})$, $SK_{ijk}=h(ID_i \| ID_{jk} \| R_c \| R_k \| M_1 \| TS_k)$, and checks $M_{27}=h(SK_{ijk} \| M_{22} \| M_{23} \| R_c \| R_k \| M_1 \| TS_k) ?= M_{25}$. If it is not equal, the session is terminated. Otherwise, PS_{jk} is authenticated by U_i. At last, U_i and PS_{jk} share the session key $SK_{ijk}=h(ID_i \| ID_{jk} \| R_c \| R_k \| M_1 \| TS_k)$.

3.4 Password and Biometric Update Phase and Dynamic Medical Server Addition Phase

Due to both of above phases have nothing with security analysis of the DAKA protocol, we will not repeat them here. For more details, please refer to [16].

4. Cryptanalysis of DAKA Protocol

In this section, we will show that the DAKA protocol is vulnerable to **internal attack**, **impers -onation attack** and **stolen smart card attack**. Moreover, DAKA protocol also cannot provide confidentiality. The details are as follows. First, we have to consider the adversary model of password and smart card based authentication protocols [22-24].

- The adversary $\mathcal{A}$ can eavesdrop, intercept, delete, and modify all messages of the common communication channel;
- $\mathcal{A}$ can obtain the secret information $\{\{ID_{S_j}, A_j, P_j\}, e_i, f_i, \tau_i, Gen(\cdot), Rep(\cdot), h(\cdot), t\}$ $(1 \leq j \leq m+m')$ in the smart card by using side-channel attacks [25-27].

4.1 Internal attacks

Assume that adversary $\mathcal{A}$ is a malicious patient with identity ID_m. Once he logged in the medical server MS_j, and obtained the **msg2=** $\{ID_{S_j}, ID_{jk}, M_{12}^m, M_{13}^m, M_{14}^m, M_{15}^m, TS_s^m\}$ on public channel at same time. Then he calculates $M_{11} = ID_m \oplus M_{12}^m$, which is static between MS_j and PS_{jk}. When a legitimate patient U_i logins the medical system, $\mathcal{A}$ can get $\{\{ID_{S_j}, \underline{A_j}, \underline{P_j}\}, e_i, f_i, \tau_i, Gen(\cdot), Rep(\cdot), h(\cdot), t\}(1 \le j \le m+m')$, **msg1=** $\{ID_{S_j}, ID_{jk}, \underline{M_3}, \underline{M_4}, M_5, TS_c\}$, **msg2=** $\{ID_{S_j}, ID_{jk}, \underline{M_{12}}, \underline{M_{13}}, \underline{M_{14}}, M_{15}, TS_s\}$, **msg3=** $\{ID_{jk}, \underline{M_{22}}, M_{23}, M_{24}, TS_k\}$; Next, $\mathcal{A}$ does the calculations as follows: $M_1 \oplus M_2 = A_j \oplus P_j$, $M_3 \oplus M_4 = M_1 \oplus M_2 \oplus R_c$, $R_c = M_3 \oplus M_4 \oplus A_j \oplus P_j$ (Here, $\mathcal{A}$ can obtains R_c), $ID_i = M_{11} \oplus M_{12}$, $R_s = M_{14} \oplus ID_i \oplus R_c$, $h(ID_i \| X_{jk}) = M_{13} \oplus R_s$, $R_k = M_{22} \oplus ID_i \oplus R_c$. Then, $\mathcal{A}$ can compute the session key $SK_{ijk} = h(ID_i \| ID_{jk} \| R_c \| R_k \| h(ID_i \| X_{jk}) \| TS_k)$ shared with U_i and PS_{jk}. Hence, the DAKA protocol suffers from internal attacks.

4.2 Impersonation attacks

Above, we know that A can obtain M_{11}, R_c. Then, $\mathcal{A}$ can initiate impersonation attacks.

Impersonation Medical server MS_j:

- $\mathcal{A}$ intercepts **msg2**. Then, $\mathcal{A}$ computes $ID_i = M_{11} \oplus M_{12}, R_s = M_{14} \oplus ID_i \oplus R_c$, $h(ID_i \| X_{jk}) = M_{13} \oplus R_s$. Next, $\mathcal{A}$ forges authentication information. $\mathcal{A}$ chooses a random number $R_{\mathcal{A}}$, and computes $M_{13} = h(ID_i \| X_{jk}) \oplus R_{\mathcal{A}}, M_{14}^{\mathcal{A}} = ID_i \oplus R_c \oplus R_{\mathcal{A}}, M_{15}^{\mathcal{A}} = h(ID_i \| M_{11} \| M_{12} \| M_{13}^{\mathcal{A}} \| M_{14}^{\mathcal{A}} \| R_c \| R_{\mathcal{A}} \| TS_s)$. Then, $\mathcal{A} \rightarrow PS_{jk}: \{ID_{S_j}, ID_{jk}, M_{12}, M_{13}^{\mathcal{A}}, M_{14}^{\mathcal{A}}, M_{15}^{\mathcal{A}}, TS_s\}$;
- On receiving the message from $\mathcal{A}$, PS_{jk} computes $M_{16} = h(ID_{S_j} \| ID_{jk} \| X_{jk})$, $M_{17} = M_{12} \oplus M_{16} = ID_i, M_{18}^{\mathcal{A}} = M_{13}^{\mathcal{A}} \oplus h(M_{17} \| X_{jk}) = R_{\mathcal{A}}$, $M_{19}^{\mathcal{A}} = M_{14}^{\mathcal{A}} \oplus M_{17} \oplus M_{18}^{\mathcal{A}} = R_c$, and checks $M_{20}^{\mathcal{A}} = h(M_{17} \| M_{16} \| M_{12} \| M_{13}^{\mathcal{A}} \| M_{14}^{\mathcal{A}} \| M_{19} \| M_{18}^{\mathcal{A}} \| TS_s) ? = M_{15}^{\mathcal{A}}$. It is easy to see that $M_{20}^{\mathcal{A}} = M_{15}^{\mathcal{A}}$, $\mathcal{A}$ is verified by the PS_{jk}.

Impersonation Physical server PS_{jk}:

- $\mathcal{A}$ intercepts **msg2**. $\mathcal{A}$ generates a random number $R_{\mathcal{A}}$, and computes $M_{22}^{\mathcal{A}} = ID_i \oplus R_c \oplus R_{\mathcal{A}}$, $M_{23}^{\mathcal{A}} = h(ID_i \| X_{jk}) \oplus R_{\mathcal{A}}$, $SK_{i_{\mathcal{A}}} = h(ID_i \| ID_{jk} \| R_c \| R_{\mathcal{A}} \| h(ID_i \| X_{jk}) \| TS_k)$, $M_{24}^{\mathcal{A}} = h(SK_{i_{\mathcal{A}}} \| M_{22}^{\mathcal{A}} \| M_{23}^{\mathcal{A}} \| R_c \| R_{\mathcal{A}} \| TS_k)$. Then, $\mathcal{A} \rightarrow U_i: \{ID_{jk}, M_{22}^{\mathcal{A}}, M_{23}^{\mathcal{A}}, M_{24}^{\mathcal{A}}, TS_k\}$;
- U_i computes $M_{25}^{\mathcal{A}} = M_{22}^{\mathcal{A}} \oplus ID_i \oplus R_c = R_{\mathcal{A}}, M_{26} = M_{23}^{\mathcal{A}} \oplus M_{25}^{\mathcal{A}} = h(ID_i \| X_{jk}), SK_{i_{\mathcal{A}}} = h(ID_i \| ID_{jk} \| R_c \| R_{\mathcal{A}} \| M_1 \| TS_k)$, and checks $M_{27}^{\mathcal{A}} = h(SK_{i_{\mathcal{A}}} \| M_{22}^{\mathcal{A}} \| M_{23}^{\mathcal{A}} \| R_c \| R_{\mathcal{A}} \| TS_k) ? = M_{25}^{\mathcal{A}}$. It is easy to see that $M_{27}^{\mathcal{A}} = M_{25}^{\mathcal{A}}$, $\mathcal{A}$ is verified by U_i.

  Hence, the DAKA protocol is vulnerable to impersonation attacks.

4.3 Stolen smart card attack

An efficient biometric based multi-server medical system must not allow an adversary $\mathcal{A}$ to misuse a user's stolen smart card to compute the session keys without knowing the user's biometric and password. In this attack, we show the DAKA protocol cannot resist the stolen smart card such that A can achieve server's secret key and previously established session key. $\mathcal{A}$ has obtained the secret information $\{\{ID_{S_j}, A_j, P_j\}, e_i, f_i, \tau_i, Gen(\cdot), Rep(\cdot), h(\cdot), t\}$ $(1 \le j \le m + m')$ in the smart card. $\mathcal{A}$ gets the previously transmitted message: **msg1**, **msg2** and **msg3**. From above, we know $\mathcal{A}$ can obtain $\{ID_i, ID_{jk}, R_c, R_k, h(ID_i \| X_{jk}), TS_k\}$. Then, $\mathcal{A}$ computes $SK_{ijk} = h(ID_i \| ID_{jk} \| R_c \| R_k \| h(ID_i \| X_{jk}) \| TS_k)$. Thus, it is clear that $\mathcal{A}$ can achieve all established session keys by using the stolen smart card.

4.4 Lack of confidentiality

According to previous analysis, we know that the session key of U_i and PS_{jk} is easily obtained by $\mathcal{A}$. $\mathcal{A}$ can also achieve all the confidential data that are transferred between the U_i and PS_{jk} using the established session key. Therefore, the DAKA protocol cannot ensure confidentiality.

5. Our Proposed Improved Protocol

To overcome the afored-discussed security flaws of the DAKA protocol, a lightweight pseudonym biometrics-based protocol (PAKA) is proposed. Our PAKA protocol is made up of six basic phases: **Registration phase**, **Login phase**, **Authentication phase**, **Password and Biometric change phase** and **Smart card upgrade phase**. The detailed steps of these phases are described as follows. The **Registration phase**, **Login phase** and **Authentication phase** are further illustrates in **Fig. 2**, **Fig. 3, Fig. 4** respectively.

5.1. Registration phase

1) **Medical Server Registration Phase:** If MS_j wants to be a legal medical server in the system. It needs to register in MRS firstly. The following steps show the detailed interactive process between MS_j and MRS, which is also shown in **Fig. 2**.

Rs1 MS_j chooses his/her ID_{S_j}. Then, $MS_j \Rightarrow MRS: ID_{S_j}$

Rs2 Upon receiving the registration message from MS_j, MRS chooses a random value β_j, computes $X_j = h(ID_{S_j} \| X_r \| \beta_j)$. Then, $MRS \Rightarrow MS_j: X_j$;

Rs3 After receiving the message X_j from MRS, MS_j calculates $BMS_j = X_j \oplus z_j$, z_j is the secret key of MS_j.

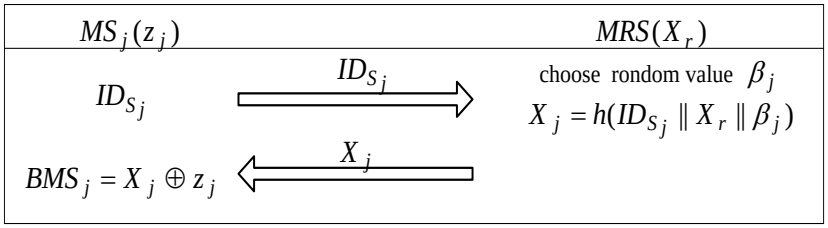

Fig. 2. Medical Server Registration Phase

2) **User Registration Phase**: When a patient U_i wants to access medical services in the system, he/she should register in MRS firstly. The following steps run between U_i and MRS as shown in **Fig. 3.**

Ru1 U_i chooses his/her ID_i, PW_i, a random number R_i, gets personal biometric data B_i, and computes biometric key σ_i by $(\sigma_i, \tau_i) = Gen(B_i)$, $RPW_i = h(ID_i \| R_i \| PW_i)$. Then, $U_i \Rightarrow MRS : ID_i, RPW_i$;

Ru2 Upon receiving the registration message from U_i, MRS chooses a random value α_i, computes $X_i, A_i, C_{ij}, D_{ij}, E_{ij}$ as **Fig. 3**, and stores $\{A_i, < ID_{S_j}, E_{ij} >, h(\cdot)\}$ in smart card,

where $j = 1,2,.....,m$. Here, m is the number of server. Then, $MRS \Rightarrow U_i$: Smart card;

Ru3 U_i computes e_i, f_i as **Fig. 3**, and adds $\{e_i, f_i, \tau_i\}$ to the smart card. Finally, $\{A_i, < ID_{S_j}, E_{ij} >, e_i, f_i, \tau_i, h(\cdot)\}$ are stored in U_i 's smart card.

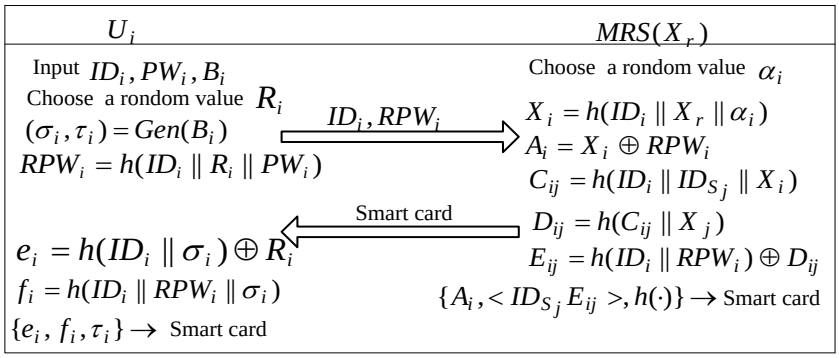

Fig. 3. Patient Registration Phase

5.2. Login phase

When U_i wants to login to MS_j to get medical services, the following operations will be performed as shown in **Fig. 4**:

L1 U_i inserts his/her smart card into the card reader, inputs his/her ID_i, PW_i and his/her biometric data B_i' read by special equipment;

L2 The smart card computes biometric key σ_i, R_i, RPW_i as **Fig.4**, and checks $f_i ? = h(ID_i \| RPW_i \| \sigma_i)$. If it does not match, the smart card terminates the session. Otherwise, the smart card generates a random value R_c and computes $X_i, C_{ij}, D_{ij}, M_1, M_2$ as **Fig. 4**.

L3 $U_i \rightarrow MS_j$: **msg1**=$\{ID_{S_j}, ID_{jk}, C_{ij}, M_1, M_2\}$;

6.3. Authentication phase

Upon receiving the login request **msg1**, MS_j performs the following operations:

V1 MS_j computes X_j, D_{ij}, R_c as **Fig. 4**. Then MS_j checks the pair (C_{ij}, R_c) according to C_{ij}

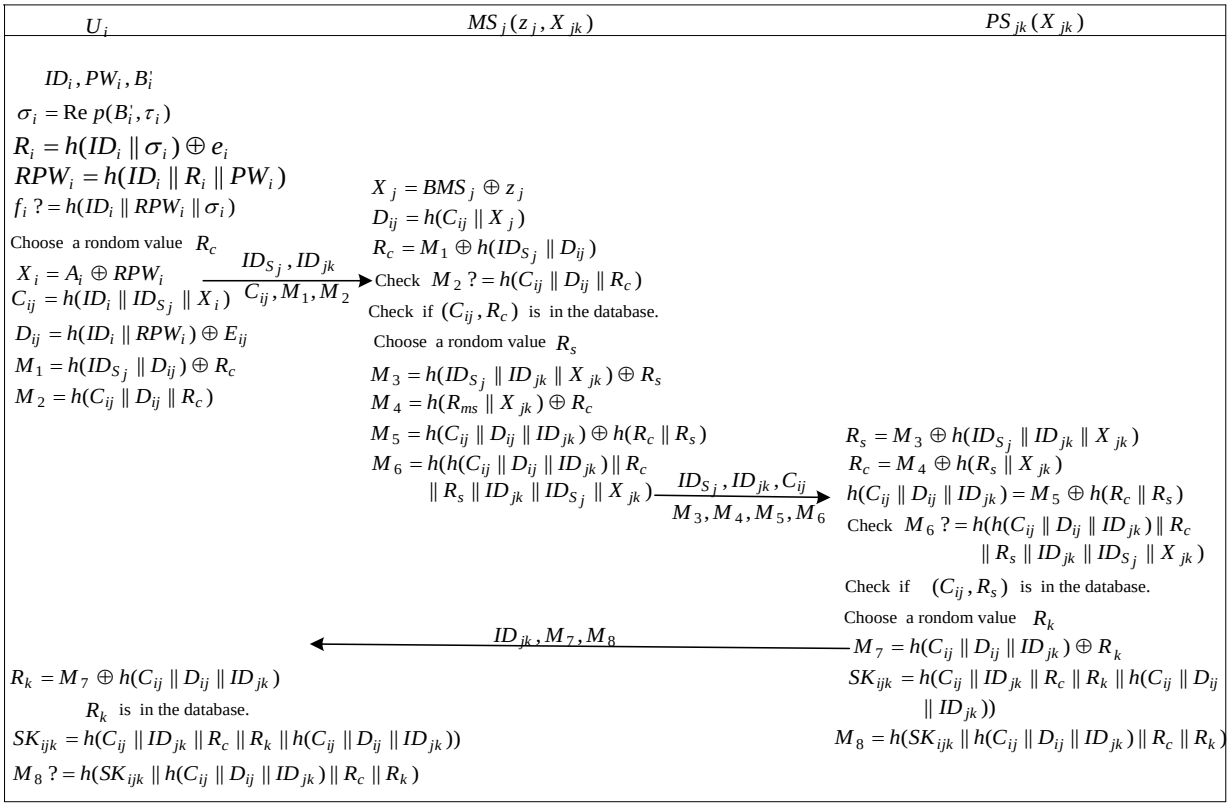

Fig. 4. Authentication phase

and $M_2 ? = h(C_{ij} \| D_{ij} \| R_c)$. If that above equality does not hold, the login request is rejected. Otherwise;

V2 MS_j replaces the R_c^{old} with R_c, stores the pair (C_{ij}, R_c) in database (Here, in order to protect the reply and man-in-the-middle attack in our protocol, MS_j stores the pair $((C_{ij}, R_c)$ in database. Meanwhile, the pair (C_{ij}, R_c) will be changed according to user's login.), chooses a random value Rs, and computes M_3, M_4, M_5, M_6 as **Fig. 4**. Then, $MS_j \rightarrow PS_{jk}$: **msg2**=$\{ID_{S_j}, ID_{jk}, C_{ij}, M_3, M_4, M_5, M_6\}$;

V3 Upon receiving **msg2**, PS_{jk} computes $R_s, R_c, h(C_{ij} \| D_{ij} \| ID_{jk})$ as **Fig. 4.** Then ,

PS_{jk} checks the pair (C_{ij}, R_s) (Here, the reason is same to MS_j), and $M_6 ?= h(M_5 \| R_c \| R_s \| ID_{jk} \| ID_{S_j} \| X_{jk})$. If the above verification fails, the login request will be rejected. Otherwise;

V4 PS_{jk} generates a random value R_k and computes M_7, SK_{ijk}, M_8 as **Fig. 4**. Then, $PS_{jk} \rightarrow U_i$:**msg3**=$\{ID_{jk}, M_7, M_8\}$;

V5 On receiving **msg3**, U_i computes R_k, SK_{ijk} as **Fig.4**, and checks $M_8 ?= h(SK_{ijk} \| h(C_{ij} \| D_{ij} \| ID_{jk}) \| R_c \| R_k)$. If it does not hold, the session is terminated.Otherwise, SK_{ijk} is valid. Meanwhile, U_i and PS_{jk} realize mutual authentication.

6.4. Password and Biometric Update Phase

It is invoked whenever U_i wants to change the old PW_i, B_i to the PW_i^{new}, B_i^{new} ; without the help of MRS :

C1 U_i inserts his/her smart card into card reader and enters ID_i , PW_i , $B_i^{'}$;

C2 The smart card computes $\sigma_i = Rep(B_i^{'}, \tau_i)$, $R_i = h(ID_i \| \sigma_i) \oplus e_i$, $RPW_i = h(ID_i \| R_i \| PW_i)$, and checks $f_i ? = h(ID_i \| RPW_i \| \sigma_i)$. If it does not match, the session is terminated Or else, U_i computes $(\sigma_i^{new}, \tau_i^{mew}) = Gen(B_i^{new})$, $RPW_i^{new} = h(ID_i \| R_i \| PW_i^{new})$, $A_i^{new} = A_i \oplus RPW_i \oplus RPW_i^{new}$, $E_{ij}^{new} = h(ID_i \| RPW_i) \oplus E_{ij} \oplus h(ID_i \| RPW_i^{new})$, $e_i^{new} = h(ID_i \| \sigma_i^{new}) \oplus K$, $f_i^{new} = h(ID_i \| RPW_i^{new} \| \sigma_i^{new})$, and updates $\{A_i, e_i, f_i, E_{ij}, \tau_i\}$ with $\{A_i^{new}, e_i^{new}, f_i^{new}, E_{ij}^{new}, \tau_i^{new}\}$ in smart card.

6.5. Smart card upgrade phase

Suppose that MRS adds some the new servers such as MS_m in the system and U_i wants to get some medical services from them. At this time, he/she needs to upgrade his/her smart card.

R1 U_i chooses his/her ID_i , PW_i , a random value R_i , gets personal biometric B_i and computes biometric key σ_i by $(\sigma_i, \tau_i) = Gen(B_i)$, $RPW_i = h(ID_i \| R_i \| PW_i)$. Then, $U_i \Rightarrow MRS : ID_i, RPW_i$;

R2 Upon receiving the registration message from U_i , MRS computes $X_i = h(ID_i \| X_r)$, $A_i = X_i \oplus RPW_i$, $C_{im} = h(ID_i \| ID_{S_m} \| X_i)$, $D_{im} = h(C_{im} \| X_{im})$, $E_{im} = h(ID_i \| RPW_i) \oplus D_{im}$, and stores $\{< ID_{S_m}, E_{im} >\}$ in smart card, where $j = 1,2,.....,m$. Here, m is the number of server. Then, $MRS \Rightarrow U_i$: Smart card;

R3 U_i computes $e_i = h(ID_i \| \sigma_i) \oplus R_i$, $f_i = h(ID_i \| RPW_i \| \sigma_i)$. Finally $\{< ID_{S_m}, E_{im} >\}$ are added in U_i 's smart card.

6. Security Analysis and Proof of PAKA Protocol

In this section, we will analyze the security of the PAKA protocol under the same adversary model mentioned in Section 4.

7.1. Security analysis

1) Patient anonymity: The PAKA protocol adopts an anonymous blind identity $C_{ij} = h(ID_i \| ID_{S_j} \| X_i)$ instead of the static identity ID_i in the public communication channel. By using a collision resistant hash function, onewayness property ensures malicious adversary $\mathcal{A}$ cannot extract the U_i 's ID_i from the eavesdropped C_{ij} . Further, in the PAKA protocol, service provider MS_j and PS_{jk} cannot know U_i 's real identity either. In this way, PAKA protocol provides patient anonymity, which can prevent the privacy leakage of patient identity.

2) Perfect forward secrecy: In our PAKA protocol, $SK_{ijk} = h(C_{ij} \| ID_{jk} \| R_c \| R_k \| h(C_{ij} \| D_{ij} \| ID_{jk}))$ is the session key shared between U_i and PS_{jk}, wherein R_c and R_k are random values chosen by U_i and PS_{jk} respectively, which are different in each session run. SK_{ijk} is hash value which cannot disclose any information. Therefore, $\mathcal{A}$ cannot infer any valuable information from the forward and backward session keys even if he gets the current session key.

3) Impersonation attack: If $\mathcal{A}$ can obtain the information $\{A_i, < ID_{S_j}, E_{ij} >, e_i, f_i, \tau_i, h(\cdot)\}$ stored in the smart card and the information **msg1**= $\{ID_{S_j}, ID_{jk}, C_{ij}, M_1, M_2\}$ **,msg2=** $\{ID_{S_j}, ID_{jk}, C_{ij}, M_3, M_4, M_5, M_6\}$ **,msg3=** $\{ID_{jk}, M_7, M_8\}$;in public channel. $\mathcal{A}$ (other medical servers, physician servers and malicious-legitimate patients) cannot get the secret information D_{ij} only shared between U_i and MS_j . So $\mathcal{A}$ cannot figure out the valid authentication message $M_2 = h(C_{ij} \| D_{ij} \| R_c)$ and $M_8 = h(SK_{ijk} \| h(C_{ij} \| D_{ij} \| ID_{jk}) \| R_c \| R_k)$ to pass the authentication. So the PAKA protocol can resist the impersonation attack.

4) Internal attacks: Assume that $\mathcal{A}$ is a malicious-legitimate patient, $\mathcal{A}$ uses his own smart card and information in public channel. He obtains nothing about other patients' secret information D_{ij} . And he also cannot get the secret information $h(ID_{S_j} \| ID_{jk} \| X_{jk})$. So he cannot succeed in forging authentication information $M_2 = h(C_{ij} \| D_{ij} \| R_c)$ and $M_8 = h(SK_{ijk} \| h(C_{ij} \| D_{ij} \| ID_{jk}) \| R_c \| R_k)$ to pass the authentication. Hence, the PAKA protocol can resist the internal attacks.

5)Password guessing attack : In our PAKA protocol, the password PW_i is involved in $A_i = X_i \oplus RPW_i$, $E_{ij} = h(ID_i \| RPW_i) \oplus D_{ij}$, $f_i = h(ID_i \| RPW_i \| \sigma_i)$, which are stored in the smart card. Assume that $\mathcal{A}$ has obtained the secret information $\{A_i, E_{ij}, f_i\}$ in the smart card using side-channel attacks [25-27]. However, guessing password PW_i without knowing the biometric key σ_i and identity ID_i is a small probability event for $\mathcal{A}$. Since biometric keys cannot be lost/forgotten, it is hard to forge and also copy [28]. Hence, $\mathcal{A}$ has no ability to derive the PW_i from $\{A_i, E_{ij}, f_i\}$. Thus, our PAKA protocol is secure against password guessing attack.

6) Stolen smart card attack: Assume that the U_i 's smart card was stolen by $\mathcal{A}$, $\mathcal{A}$ obtained the secret information $\{A_i, < ID_{S_j}, E_{ij} >, e_i, f_i, \tau_i, h(\cdot)\}$, where $A_i = X_i \oplus RPW_i$, $e_i = h(ID_i \| \sigma_i) \oplus R_i$, $E_{ij} = h(ID_i \| RPW_i) \oplus D_{ij}$, $f_i = h(ID_i \| RPW_i \| \sigma_i)$. However, X_i, X_j, RPW_i, ID_i

and D_{ij} are unknown to $\mathcal{A}$ and protected by onewayness hash function, $\mathcal{A}$ has no way to guess the X_i, X_j, RPW_i, ID_i and D_{ij} at the same time. Therefore, $\mathcal{A}$ cannot update the password of U_i. Besides, since X_i, X_j, RPW_i, ID_i and D_{ij} are unknown to $\mathcal{A}$, $\mathcal{A}$ cannot forge the valid login request **msg1** by using the stolen smart card. Hence, our PAKA protocol is free from the stolen smart card attack.

7) **Replay attack:** Suppose $\mathcal{A}$ intercepts the massage **msg1**, where $C_{ij} = h(ID_i \| ID_{S_j} \| X_i)$, $M_1 = h(ID_{S_j} \| D_{ij}) \oplus R_c$, $M_2 = h(C_{ij} \| D_{ij} \| R_c)$, and replies this message to MS_j. However, MS_j stores the pair (C_{ij}, R_c) in its database. Later, when MS_j receives the next login request message **msg1,** MS_j computes $D_{ij} = h(C_{ij} \| X_j)$, $R_c = M_1 \oplus h(ID_{S_j} \| D_{ij})$, and compares R_c corresponding to C_{ij}. If it matches, MS_j ensures that this request message is a replay message and rejects this request. Or else, MS_j replaces R_c with R_c^{new}. So does the PS_{jk}. Hence, our PAKA protocol can resist the replay attack.

8) **Man-in-the-middle attack:** In this attack, $\mathcal{A}$ may try to impersonate a valid patient U_i, or a medical server MS_j, or a physician server PS_{jk} by intercepting the message. However, in the PAKA protocol the secret value D_{ij} is only shared between U_i and MS_j, it will never be discovered by anybody else except U_i and MS_j. PS_{jk} only knows the $h(C_{ij} \| D_{ij} \| ID_{jk})$. Hence, our PAKA protocol is secure against man-in-the-middle attack.

7.2. Security proof

In this section, we will prove the PAKA protocol can provide secure authentication and key agreement by using the widely-accepted BAN logic [10], [11], [29].

The notations and rules about BAN logic are illustrated as follows:

$\#(X)$: X is fresh.

$P \Rightarrow X$: P has jurisdiction over X.

$P \lhd X$: P sees X.

$P \equiv X$: P believes X is true.

$P \sim X$: P once said X.

$< X >_Y$: X is combined with Y.

(X, Y): X or Y is one part of (X, Y).

$P \underline{\underline{X}} Q$: X is secretly known to P and Q and trusted by them.

$P \overset{k}{\longleftrightarrow} Q$: P and Q may use the shared key k to communicate. The key *k* will never be discovered by anyentity except P and Q.

- **Rule1 :** The message-meaning rule: $\dfrac{P \equiv P \underline{\underline{Y}} Q, P \lhd < X >_Y}{P \equiv Q \sim X}$;
- **Rule2 :** The nonce-verification rule: $\dfrac{P \equiv \#(X), P \equiv Q \sim X}{P \equiv Q \equiv X}$;

- **Rule3 :** The jurisdiction rule: $\dfrac{P \mid\equiv Q \mid\Rightarrow X, P \mid\equiv Q \mid\equiv X}{P \mid\equiv X}$;
- **Rule4 :** The freshness rule: $\dfrac{P \mid\equiv \#(X)}{P \mid\equiv \#(X,Y)}$.

According to the analytic procedures of the BAN logic, the PAKA protocol should achieve the following goals:

- **Goal1:** $U_i \mid\equiv PS_{jk} \mid\equiv (U_i \xleftrightarrow{SK} PS_{jk})$;
- **Goal2:** $U_i \mid\equiv (U_i \xleftrightarrow{SK} PS_{ij})$;
- **Goal3:** $PS_{jk} \mid\equiv U_i \mid\equiv (U_i \xleftrightarrow{SK} PS_{jk})$;
- **Goal4:** $PS_{jk} \mid\equiv (U_i \xleftrightarrow{SK} PS_{jk})$.

First, we idealize the communication messages of the PAKA protocol as follows: (In order to simplify, let $A = h(C_{ij} \| D_{ij} \| ID_{jk})$.

- **msg1:** $U_i \rightarrow MS_j :< C_{ij}, ID_{S_j}, ID_{jk}, R_c >_{U_i \underline{\underline{D_{ij}}} MS_j}$;
- **msg2:** $MS_j \rightarrow PS_{jk} :< ID_{S_j}, ID_{jk}, R_c, R_s, U_i \underline{\underline{A}}(MS_j, PS_{jk}) >_{MS_j \underline{\underline{X_{ij}}} PS_{jk}}$;
- **msg3:** $PS_{jk} \rightarrow U_i :< ID_{jk}, R_c, R_k, U_i \xleftrightarrow{SK} PS_{jk} >_{U_i \underline{\underline{A}}(MS_j, PS_{jk})}$.

.

Second, the following assumptions about the initial state are made to analyze the PAKA protocol:

- **H1:** $U_i \mid\equiv \#(R_c)$;
- **H2:** $MS_j \mid\equiv \#(R_s)$;
- **H3:** $PS_{jk} \mid\equiv \#(R_k)$;
- **H4:** $U_i \mid\equiv U_i \underline{\underline{A}}(MS_j, PS_{jk})$;
- **H5:** $U_i \mid\equiv PS_{jk} \mid\Rightarrow (U_i \xleftrightarrow{SK} PS_{jk})$;
- **H6:** $MS_j \mid\equiv U_i \overset{D_{ij}}{=\!=} MS_j$;
- **H7:** $PS_{jk} \mid\equiv PS_{jk} \underline{\underline{X_{ij}}} MS_j$
- **H8:** $PS_{jk} \mid\equiv MS_j \mid\Rightarrow (U_i \underline{\underline{A}}(MS_j, PS_{jk}))$;
- **H9:** $PS_{jk} \mid\equiv U_i \mid\Rightarrow (U_i \xleftrightarrow{SK} PS_{jk})$.

Third, the main proofs of the idealized form of PAKA protocol based on the BAN logic rules and assumptions is analyzed as follows:

From **msg3**, we get:

S1: $U_i \lhd< ID_{jk}, R_c, R_k, U_i \xleftrightarrow{SK} PS_{jk} >_{U_i \underline{\underline{A}}(MS_j, PS_{jk})}$;

From **H4**, **S1** and **Rule1**, we get:

S2: $\dfrac{U_i \mid\equiv U_i \underline{\underline{A}}(MS_j, PS_{jk}), U_i \lhd< ID_{jk}, R_c, R_k, U_i \xleftrightarrow{SK} PS_{jk} >_{U_i \underline{\underline{A}}(MS_j, PS_{jk})}}{U_i \mid\equiv PS_{jk} \mid\sim< ID_{jk}, R_c, R_k, U_i \xleftrightarrow{SK} PS_{jk} >}$;

From **H1**, **S2**, **Rule2** and Rule4 we have:

S3: $\dfrac{U_i \mid\equiv \#(R_c)}{U_i \mid\equiv \#< ID_{jk}, R_c, R_k, U_i \xleftrightarrow{SK} PS_{jk} >}$;

$$\frac{U_i \mid\equiv \#<ID_{jk}, R_c, R_k, U_i \xleftrightarrow{SK} PS_{jk}>, U_i \mid\equiv PS_{jk} \mid\sim<ID_{jk}, R_c, R_k, U_i \xleftrightarrow{SK} PS_{jk}>}{U_i \mid\equiv PS_{jk} \mid\equiv<ID_{jk}, R_c, R_k, U_i \xleftrightarrow{SK} PS_{jk}>};$$

$$U_i \mid\equiv PS_{jk} \mid\equiv (U_i \xleftrightarrow{SK} PS_{jk})\ (\textbf{Goal1});$$

From **H5**, **S3**, and **Rule3** we obtain:

S4: $$\frac{U_i \mid\equiv PS_{jk} \mid\Rightarrow (U_i \xleftrightarrow{SK} PS_{jk}), U_i \mid\equiv PS_{jk} \mid\equiv (U_i \xleftrightarrow{SK} PS_{jk})}{U_i \mid\equiv (U_i \xleftrightarrow{SK} PS_{jk})};(\textbf{Goal2})$$

From **msg1**, we get:

S5: $MS_j \triangleright\!<C_{ij}, ID_{S_j}, ID_{jk}, R_c>_{U_i \underline{\underline{D_{ij}}} MS_j}$;

From **H6**, **S5** and **Rule1**, we also get:

S6: $$\frac{MS_j \mid\equiv U_i \overset{D_{ij}}{=\!=} MS_j, MS_j \triangleright\!<C_{ij}, ID_{S_j}, ID_{jk}, R_c>_{U_i \underline{\underline{D_{ij}}} MS_j}}{MS_j \mid\equiv U_i \mid\sim<C_{ij}, ID_{S_j}, ID_{jk}, R_c>};$$

Here, we know that $MS_j \mid\equiv \#(R_c)$, and MS_j shares R_c with PS_{jk}. Then, PS_{jk} $PS_{jk} \mid\equiv \#(R_c)$.

From **msg2**, we get:

S7: $PS_{jk} \triangleleft\!<ID_{S_j}, ID_{jk}, R_c, R_s, U_i \underline{\underline{A}}(MS_j, PS_{jk})>_{MS_j \underline{\underline{X_{ij}}} PS_{jk}}$;

From **H7**, **S7** and **Rule1**, we also get:

S8: $$\frac{PS_{jk} \mid\equiv PS_{jk} \underline{\underline{X_{ij}}} MS_j, PS_{jk} \triangleleft\!<ID_{S_j}, ID_{jk}, R_c, R_s, U_i \underline{\underline{A}}(MS_j, PS_{jk})>_{MS_j \underline{\underline{X_{ij}}} PS_{jk}}}{PS_{jk} \mid\equiv MS_j \mid\sim<ID_{S_j}, ID_{jk}, R_c, R_s, U_i \underline{\underline{A}}(MS_j, PS_{jk})>};$$

From **S6**, **S8**, **Rule2** and **Rule4** we also have:

S9: $$\frac{PS_{jk} \mid\equiv \#(R_c)}{PS_{jk} \mid\equiv \#<ID_{S_j}, ID_{jk}, R_c, R_s, U_i \underline{\underline{A}}(MS_j, PS_{jk})>}$$

$$\frac{PS_{jk} \mid\equiv \#<ID_{S_j}, ID_{jk}, R_c, R_s, U_i \underline{\underline{A}}(MS_j, PS_{jk})>, PS_{jk} \mid\equiv MS_j \mid\sim<ID_{S_j}, ID_{jk}, R_c, R_s, U_i \underline{\underline{A}}(MS_j, PS_{jk})>}{PS_{jk} \mid\equiv MS_j \mid\equiv<ID_{S_j}, ID_{jk}, R_c, R_s, U_i \underline{\underline{A}}(MS_j, PS_{jk})>};$$

$$PS_{jk} \mid\equiv MS_j \mid\equiv (U_i \underline{\underline{A}}(MS_j, PS_{jk}))$$

From **H8**, **S9** and **Rule3** we also get:

S10: $$\frac{PS_{jk} \mid\equiv MS_j \mid\Rightarrow (U_i \underline{\underline{A}}(MS_j, PS_{jk})), PS_{jk} \mid\equiv MS_j \mid\equiv (U_i \underline{\underline{A}}(MS_j, PS_{jk}))}{PS_{jk} \mid\equiv (U_i \underline{\underline{A}}(MS_j, PS_{jk}))};$$

From **H3**, **S6**, and **S10** we can obtain:

S11: $PS_{jk} \mid\equiv U_i \mid\equiv (U_i \xleftrightarrow{SK} PS_{jk})$;(**Goal3**)

From **H9**, **S11**, and **Rule3** we also obtain:

S12: $$\frac{PS_{jk} \mid\equiv U_i \mid\Rightarrow (U_i \xleftrightarrow{SK} PS_{jk}), PS_{jk} \mid\equiv U_i \mid\equiv (U_i \xleftrightarrow{SK} PS_{jk})}{PS_{jk} \mid\equiv (U_i \xleftrightarrow{SK} PS_{jk})}.(\textbf{Goal4})$$

According to **Goal1**, **Goal2**, **Goal3** and **Goal4**, we can conclude that our PAKA protocol is truly able to achieve the scheduled security goals.

7. Performance Evaluation

Table 2. Performance comparison among relevant authentication protocols

	Amin[15]	DAKA[16]	PAKA
patient computation cost	6H	6H	10H
medical server computation cost	6H	6H	7H
physician server computation cost	6H	6H	7H
communication cost/bit	2400	2880	2400
storage overhead/bit	480*m+160	480*m+160	320*m+640

In this section, the performance and security features of the PAKA protocol with other related protocols are given. The results are depicted in **Table 2** and **Table 3**. Let h denote hash function operation and th be the time complexity for hash function operation. Since the time of concatenation operation and XOR operation are negligible as compared to the other time-consuming operations, we do not take them into account. The time of a fuzzy extractor operation is the same among these protocols, we also do not take them into account. Based on the results in [30], $t_h \approx 0.0023\,ms$. Amin's protocol and the DAKA protocol need 0.0414ms and 0.0414ms, respectively. Our PAKA needs 0.0552ms, which slightly increase.

Table 3. Security features comparison among relevant authentication protocols

	Amin[15]	DAKA[16]	PAKA
User anonymity	No	No	Yes
Forward secrecy	No	No	Yes
Session key agreement	Yes	Yes	Yes
Resistance to off-line password guessing attack	No	Yes	Yes
Resistance to key compromise impersonation attack	No	No	Yes
Resistance to man-in-middle attack	No	No	Yes
Provable security	No	Yes	Yes

But it is much smaller than other AKA protocols which are based on public key encryption algorithms. Without loss of generality, assume that random values, times-stamps, the outputs of hash function and encryption is 160 bits [10]. **Fig. 5** roughly shows the storage overhead of Amin, DAKA, and PAKA protocols. m is the number of Medical servers. $320m + 640 \le 480m + 160$ holds when $m \ge 3$. Obviously, the number of servers in a multi-server medical system is easy to more than 3. With the increase of m, PAKA need fewer storage space than protocols [15], [16]. For the 8-bit microcontroller platform, the cost of receiving one byte is 28.6 uJ, which is roughly half of that reqired to transmit a byte (59.2uJ) [31-32], and one hash operation of SHA-1 is 5.9 uJ/byte. In patients' side, Amin et al's protocol needs 7.104 mJ, 1.716 mJ and 0.708 mJ for transmit, receive and hash operation respectively, and the DAKA protocol needs 7.104 mJ, 2.860 mJ and 0.708 mJ, respectively. The energy the PAKA needs is 5.920 mJ, 1.716 mJ and 1.180 mJ, respectively. A mobile user consumes 6.942 mJ to complete a login and AKA process. Even with 1% of energy available from a miniature 100 mAh battery, the device still can perform about 1900 PAKA handshakes.

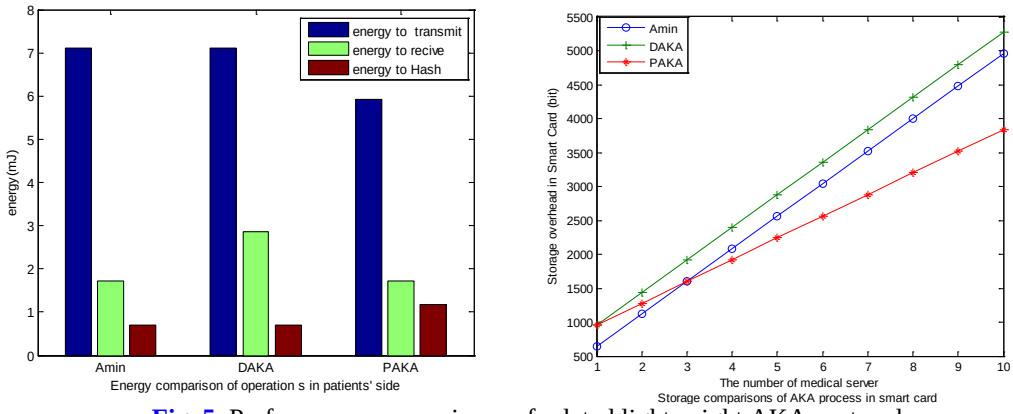

Fig. 5. Performance comparisons of related lightweight AKA protocol

8. Conclusion and Ongoing Work

System security and patients privacy-preserved are a challenging issue in distributed medical authentication systems. A lightweight pseudonym authentication and key agreement protocol (PAKA) for multi-medical server architecture in TMIS presented in this paper is trying to find a balance between the system security and patients privacy-preserved. The PAKA protocol investigates a systematic approach of multi-factor authentication: password, smart card, biological key. Only the register center MRS know patient's identity, it not only realizes anonymity to protect patient's privacy, but also addresses other prominent issues (e.g. error-tolerance). Meanwhile the PAKA protocol is proven secure by the BAN logic. Compared with the recently relevant schemes, the PAKA protocol has better performance (lightweight and energy-saving) and better security features. Thus, PAKA protocol is more secure and efficient for computation-limited mobile device. The future work is to fully identify the practical threats on multi-factor AKA protocols and develop concrete RFID based multi-factor AKA protocols in multi-medical server environment with better performance and wireless body area networks (WBAN).

References

[1] D. Florencio and C. Herley, "A large-scale study of web password habits," in *Proc. of the 16th International Conference on World Wide Web*. pp. 657-666, May 8-12, 2007. Article (CrossRef Link)

[2] Z. Y. Wu, Y. C. Lee, F. P. Lai, et al., "A secure authentication scheme for telecare medicine information systems," *Journal of Medical Systems*, vol.36, no.3, pp.1529-1535, 2012. Article (CrossRef Link)

[3] D. He, J .Cao and R .Zhang, "A more secure authentication scheme for telecare medicine information systems," *Journal of Medical Systems*, vol.36, no.3, pp.1989-1995, 2012. Article (CrossRef Link)

[4] J. Wei, X. Hu and W. Liu, "An improved authentication scheme for telecare medicine information systems," *Journal of Medical Systems*, vol. 36, no.6, pp. 3597-3604, 2012. Article (CrossRef Link)

[5] Z. Zhu, "An efficient authentication scheme for telecare medicine information systems," *Journal of Medical Systems*, vol. 36, no.6, pp. 3833-3838, 2012. Article (CrossRef Link)

[6] T.F. Lee, I.P. Chang, T. H. Lin, et al., "A secure and efficient password-based user authentication scheme using smart cards for the integrated EPR information system," *Journal of Medical Systems*, vol. 37, no.3, pp. 3867-3872, 2012. Article (CrossRef Link)
[7] Z. Tan, "An efficient biometrics-based authentication scheme for telecare medicine information systems," *Przeglad Elektrotechniczny*, vol. 89, no.5, pp.200-204, 2013. Article (CrossRef Link)
[8] X. Yan, W. Li, P. Li, et al., "A secure biometrics-based authentication scheme for telecare medicine information systems," *Journal of Medical Systems*, vol. 37, no.5, pp. 1-6, 2013. Article (CrossRef Link)
[9] D. Mishra, A. Das and S. Mukhopadhyay, "A secure user anonymity preserving biometric-based multi-server authenticated key agreement scheme using smart cards," *Expert Systems with Applications*, vol. 41, no.18, pp. 8129-8143, 2014. Article (CrossRef Link)
[10] D. He, D. Wang, "Robust biometrics-based authentication scheme for multiserver environment," *IEEE Systems Journal*, vol. 9, no.3, pp. 816-823, 2015. Article (CrossRef Link)
[11] V. Odelu, A. Das and A. Goswami, "A secure biometrics-based multi-server authentication protocol using smart cards," *IEEE Transactions on Information Forensics and Security*, vol. 10, no.9, pp. 1953-1966,2015. Article (CrossRef Link)
[12] A. Reddy, A. Das, E. Yoon, et al., "An anonymous authentication with key-agreement protocol for multi-Server architecture based on biometrics and smartcards," *KSII Transactions on Internet & Information Systems*, vol. 10, no.7, pp. 3371-3396, 2016.Article (CrossRef Link)
[13] Lee, Hanwook, et al., "Forward anonymity-preserving secure remote authentication scheme," *KSII Transactions on Internet & Information Systems* ,vol. 10, no.3,pp. 1289-1310, 2016. Article (CrossRef Link)
[14]Y. Lu, et al., "Robust ID-based mutual authentication and key agreement scheme preserving user anonymity in mobile networks," *KSII Transactions on Internet & Information Systems*, vol. 10, no.3, pp. 1273-1288, March 31, 2016. Article (CrossRef Link)
[15] R. Amin and G. 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, 2015. Article (CrossRef Link)
[16] A. Das, V. Odelu and A. Goswami, "A secure and robust user authenticated key agreement scheme for hierarchical multi-medical server environment in TMIS," *Journal of Medical Systems*, vol. 39, no.9, pp. 1-24, 2015.Article (CrossRef Link)
[17] E. Dawson, J. Lopez, et al., "BAAI: Biometric authentication and authorization infrastructure," in *Proc. of IEEE Int. Conf. on Information Technology: Research and Education (ITRE)*, pp. 371-382, Aug.11-23, 2003. Article (CrossRef Link)
[18] X, Li , J, Niu, K ,M.K ,et al., " Robust biometrics based three-factor remote user authentication scheme with key agreement," in *Proc. of IEEE Int. Symp. Biometr. Security Technologies*, pp. 105-110, July 2-5, 2013. Article (CrossRef Link)
[19] A. Makrushin, T. Scheidat and C. Vielhauer, "Improving reliability of biometric hash generation through the selection of dynamic handwriting features," *Transactions on Data Hiding and Multimedia Security VIII*. Springer Berlin Heidelberg, pp. 19-41, 2012. Article (CrossRef Link)
[20] Q. Zhang, Y. Yin, et al., "A novel serial multimodal biometrics framework based on semi-supervised learning techniques," in *Proc. of IEEE Trans. Inf. Forensics Security*, vol. 9, no.10, pp. 1681-1694, 2014. Article (CrossRef Link)
[21] M. A. Pathak, B. Raj, S. D. Rane et al., "Privacy-preserving speech processing: cryptographic and string-matching frameworks show promise," *IEEE Signal Process Magazine*, pp. 62-74, vol. 30, no.2, 2013. Article (CrossRef Link)
[22] Y. Wang, "Password protected smart card and memory stick authentication against off-line dictionary attacks," in *Proc. of 27th Information Security and Privacy Conference,Greece*, pp. 489-500, June 4-6,2012. Article (CrossRef Link)
[23]D. He, D. Wang."Robust biometrics-based authentication scheme for multi-server environment," *IEEE Systems Journal*, vol. 9, no. 3, pp. 816-823, 2015. Article (CrossRef Link)

[24]D. He, S. Zeadally, N.Kumar, et al., "Anonymou s authentication for wireless body area networks with provable security," IEEE *Systems Journal* , vol.22, no.8, pp.1-12, 2016. Article (CrossRef Link)

[25]D. He, N. Kumar, H. Shen, et al., "One-to-many authentication for access control in mobile pay-TV Systems," *Science China-Information Sciences*, vol. 59, no. 5, pp. 1–14, 2016. Article (CrossRef Link)

[26] R. Pippal, C. Jaidhar and S. Tapaswi, "Robust smart card authentication scheme for multi-server architecture," *Wireless Personal Communications*, vol.72, no.1, pp.729-745, 2013. Article (CrossRef Link)

[27] N. Huyen,M. Jo,T. Nguyen,et al., "A beneficial analysis of deployment knowledge for key distribu -tion in wireless sensor networks," *Security and Communication Networks*, vol.5, no.5 pp.485-495, 2012. Article (CrossRef Link)

[28] N. Zhang, Y. Zang and J. Tian, "The integration of biometrics cryptography —A new solution for secure identity authentication," *Journal of Cryptologic Research*, vol.2, no.2, pp.156-176, 2015.Article (CrossRef Link)

[29] X. Li, J. Niu, S. Kumari. et al., "An enhancement of a smart card authentication scheme for multi-server architecture," *Wireless Personal Communications*, vol.80, no.1, pp.175-192, 2015. Article (CrossRef Link)

[30] H. Kilinc and T. Yanik, "A survey of SIP authentication and key agreement schemes," *IEEE Communications Surveys & Tutorials*, vol.16, no.2, pp.1005-1023, 2014. Article (CrossRef Link)

[31] A. S. Wander, N. Gura, H. Eberle, et al., "Energy analysis of public-key cryptography for wireless sensor networks," in *Proc. of 3rd IEEE International Conference on Pervasive Computing and Communications*, pp.324–328,March 8-12, 2005. Article (CrossRef Link)

[32] Y. Li, W. Chen, Z. Cai, et al., "CAKA: A novel certificateless-based cross domain authenticated key agreement protocol for wireless mesh networks," *Wireless Networks*, vol.22, no.8, pp.2523–2535, 2016. Article (CrossRef Link)

Xiaoxue Liu received her B. S. degree from Bohai Univ. in 2014. She now is a M.S. degree candidate in Applied Mathematics with the School of Mathematics and Information Science, Shaanxi Normal University, Xi'an, China. Her research interests include security protocols and its analysis.

Yanping Li received her M. S. degree from Shaanxi Normal University in 2004 and Ph. D degree from Xidian University in 2009, Xi'an, China. She now is an associate professor with the School of Mathematics and Information Science, Shaanxi Normal University. Her research interests include public key cryptography and its applications

Juan Qu received the M.S. degree in Applied Mathematics from Shaanxi Normal University in 2009.She currently is an associate professor at School of Mathematics and Statistics, Chongqing Three Gorges University. Her research interests include security protocols and its security analysis.

Yong Ding received the B.S. degree in Mathematics from Sichuan University in 1998 and his M.S. degree and Ph.D degree in Cryptography from Xidian University in 2002, 2005, respectively. He now is a professor in Guilin University of Electronic Technology, China. His current research interests include cryptography and information security.