Using Dynamic Passwords for the Exchange and Sharing of Personal Health Records: A Reliable User Authentication Scheme

Wun-Lin Chen[1], Tias Kurniati[2], Zhen-Yu Wu[3], Yu-Min Huang[2], Sheng-Der Hsu[1]

[1] Division of Traumatology, Department of Surgery, Tri-Service General Hospital, National Defense Medical Center, Taiwan
[2] Department of Statistics, Tunghai University, Taiwan
[3] Department of Information Management, National Penghu University of Science and Technology, Taiwan
m89010713@mail.ndmctsgh.edu.tw, d07470701@thu.edu.tw, zywu@gms.npu.edu.tw, yumin@thu.edu.tw, f1233j@yahoo.com.tw

Abstract

The personal health records (PHRs) is a patient-centered information exchange model that allows people to autonomously maintain and manage their own personal records, including access and share their lifelong health information. A method must be implemented to protect PHRs on unsecured network and to prevent unauthorized users from accessing and modifying the PHRs during data transmission with the servers. User authentication protocols should be able to ensure the safety of user communications and data transmission on unsecured networks. Password-based user authentication is the most widely used among the currently available authentication mechanisms because of its convenience and efficiency. A password mechanism offers advantages because of its simplicity and the dependence on human's memories. On the other hand, it is easily cracked by brute force attacks such offline guessing attacks or spoofing attack and impersonation problems that may occur when the password is hacked. Therefore, this study aimed to investigate the usage of a dynamic password-based user authentication scheme on PHRs in which the characteristics of a dynamic password would prevent attackers from intercepting the correct password or guessing a user's password. Additionally, the scheme developed in this study can also resist common attacks such as replay attacks, stolen-verifier attacks, server spoofing attacks, and impersonation attacks, among others.

Keywords: PHRs, Data transmission, Password-based user authentication, Off-line guessing attacks, Dynamic password

1 Introduction

Personal health records (PHRs) have become a major model of patient-oriented medical information exchange. PHRs are stored in a centralized data center of a service provider and accessed throughout a network. Users can access their PHR service providers anywhere and create, manage, and control their PHRs at any time via web browsers and the Internet. The PHR model enables the relatively efficient storing, access, and sharing of medical information. In particular, the users have full control over their medical records and can effectively share their data with other users such as medical institutions, health insurance providers, family members, or friends. PHRs can improve the accuracy and quality of personal health care and lower health care costs.

Since the advent of cloud computing, the majority of suppliers of health care information technology and healthcare providers have begun to transfer their PHR services to the cloud. Software service providers can provide cloud storage space and software as a service (SaaS) with almost unlimited and flexible storage as well as computational resources [1]. To reduce operating costs, an increasing number of PHR providers have moved their PHR application services and data storage to the cloud instead of building dedicated data centers. For example, Google and Microsoft are two major cloud providers which offer cloud-based PHR services, namely Google Health1 and Microsoft HealthVault. Investment in PHRs has been typically motivated by the goals of attaining profits and efficiency, increasing patients' rights or improving disease management. Nevertheless, patients are primarily concerned with the security and confidentiality of their PHRs and other healthcare systems. In the United States, the Health Insurance Portability and Accountability Act (HIPAA) of 1996 [1] provided legal protection for the privacy and security of PHRs; however, it only applied to covered entities such as health plans, healthcare clearinghouses and healthcare providers. At the time, the legislation did not contemplate the emergence of cloud-based PHR service providers such as Dossia, Microsoft and Google; therefore, these service providers were not included as covered entities.

*Corresponding Author: Tias Kurniati; E-mail: d07470701@thu.edu.tw
DOI: 10.3966/160792642020072104014

Healthcare organizations (HCOs) and e-health services covered by HIPAA encounter the problem of implementing effective and cost-efficient security and privacy policies and remaining compliant with HIPAA regulations. Under HIPAA, HCOs must implement comprehensive policies, standards, guidelines and procedures to securely maintain their medical information. Their security and privacy policies apply to PHRs, including electronic medical records (EMR) and electronic health records (EHR) [1]. Third-party businesses that provide PHR solutions are not subject to HIPAA regulations, but ensuring the security and privacy of PHRs is a critical issue for these businesses as well as patients.

Storing PHRs in the cloud securely requires careful evaluation of privacy and system security. In comparison with traditional paper-based medical records, PHRs can be secured with additional features such as passwords and record tracking. However, patients may lose some actual control of their medical data when PHRs are moved to a cloud server. Furthermore, storage in the cloud renders PHRs vulnerable to various threats especially when HIPAA has yet to establish adequate regulations for safeguarding PHRs against threats from cloud computing. The threats include a lack of strict verification of user identity, insecure authentication and authorization of user interfaces, an abuse of cloud computing for illegal behavior, ill-intentioned employees of cloud service providers, problems associated with the sharing of space, and stolen data or services. Because of these threats, this study concluded that additional steps must be implemented to verify user identity and validate that sensitive patient data can be safely stored in a cloud server [2].

A few medical systems have already cooperated with telecommunication companies to introduce cloud technology into medical application services. These include cloud-based electronic medical records, cloud-based nursing information systems, and hospital information system clouds. Various hospital branches have introduced private medical cloud programs to provide patients with superior real-time medical care services, accordingly increase service quality and operational efficiency of these medical institutions. Combining PHRs and cloud services yields numerous benefits:

(1) Reduced cost: Because cloud service providers provide an infrastructure, a platform, software and storage space, hospitals are not required to establish their own dedicated medical data centers, which reduce the cost of building and updating hardware and software facilities. Regarding platform as a service (PaaS) and SaaS, cloud service providers employ various information technology (IT) professionals who are responsible for developing each service. Because hospitals can select the most imperative services and forgo others, they can reduce their IT costs as well as the cost associated with human resources management.

(2) Resource sharing and exchange: Cloud computing technology is based on the Internet. Files from various sources in the cloud can be accessed via the Internet, which enables the rapid sharing and exchange of medical information at all times.

(3) Dynamic scalability of resources: PHR systems must be scaled up constantly to serve large numbers of users. A scalable system should be capable of supporting considerable growth of the number of users. Cloud services are extremely flexible in terms of vertical scaling (i.e., up and down) and can satisfy hospitals' demand for the expansion of medical information systems and storage.

(4) On-demand self-service: Cloud computing consists of a shared pool of resources (e.g., networks, servers, storage, applications, and services), which rapidly provide dynamic configuration. Depending on their needs, hospitals can purchase appropriate configurations from cloud service providers. When multiple users submit requests, the cloud optimizes resource utilization and configures services and storage for users flexibly.

(5) Increased flexibility: Authorized users may access medical files stored in the cloud at all times. Furthermore, edits made to a temporary copy of a file by any user are automatically updated to the permanently stored file. The cloud is thus convenient for accessing data, integrating medical records, and sharing of medical resources without spatial and temporal limitations.

(6) Elimination of restrictions to particular devices: Users may receive services through any type of computer or portable device (e.g., smartphones and various types of laptop computers) as long as they have a connection to the Internet. The elimination is advantageous to facilitate the use of health management services such as blood pressure monitors that are available on devices.

(7) High scalability and service integration: Cloud computing can integrate information and services such as health education, health management, drug safety information, exercise programs, and dietary intake analysis from various providers. In addition, cloud computing data centers can store diverse data for management and analysis for various purposes such as medical research. Interhospital medical services and value-added information services for patients (e.g., telecare and family physicians) can also be integrated and extended through the cloud.

Cloud-based PHRs can be effectively used to share medical information, which prevents the waste of medical resources and provides patients with the right to fully control their own medical records [2]. For hospitals, the cost of establishing PHRs in the cloud is considerably lower than that of building their own data centers. The infrastructure as a service, PaaS and SaaS offered by cloud service providers reduce the burden

on hospital management, allowing a hospital to focus on delivering high-quality medical services.

Regarding the security of the cloud computing environment, the security mechanisms of the information system should be strengthened to effectively ensure the confidentiality and the authorized access to PHRs. To address the risk of the potential exposure of confidential patient data, PHR service providers should encrypt patient data as well as provide patients with fully control over the medical records they elect to share.

To solve this kind of problem, relevant user authentication schemes and secret-key distribution protocols have been proposed [3-6]. Among these protocols, the password-based mechanism is the most widely utilized method because of its efficiency. Under such mechanism, each user is allowed to select and keep a password in mind, without the need for any additional assistant devices for user authentication.

Unfortunately, most of this type of schemes is proven to be unable to resist off-line password guessing attacks [3, 5-8]. Adversary can correctly crack the password of a specific user by brute force attacks through intercepted information or self-generated parameters. Endless possible problems are then presented with the hacking of the password. For example, the malicious attacker may masquerade as a server to communicate with other users or impersonate as the user to log into a server to acquire services.

Therefore, a new user authentication scheme with dynamic passwords is proposed. While the scheme requires authentication, with dynamic passwords, the attacker will fail to accurately guess the password the user uses each time. Not only password guessing attacks and various common attacks like replay attacks, stolen-verifier attacks, server spoofing attacks, impersonation attacks, denial of service attacks on authentication mechanisms can be resisted, but also the perfect forward secrecy. The concept of this proposal is similar to one-time password user authentications, yet it does not have security problems or weaknesses appeared in these schemes [9-11].

Especially, this scheme is suitable for the environments requiring little verification that there is always a new password after the authentication. For example, a medical environment with great confidentiality is required for hospitals sharing or exchanging information within the hospital systems or among hospitals. The proposed scheme is appropriate in such environments since the authentication and the communications are not always required, perhaps only once a month or a season.

The rest of this paper is organized as follows. Section 2 introduces the related works to this proposal. Section 3 illustrates the proposed user authentication scheme with dynamic passwords. Following, security analyses are done in Section 4, and conclusions are drawn in Section 5.

2 Related Works

Password authentication is regarded as one of the simple and convenient authentication schemes for legal users to utilize the resources of remote systems over insecure networks. It has been applied to many Internet systems, including remote login, government organizations, private corporations, and database management systems.

The first remote user authentication scheme based on the concept of the password-based technology was proposed by Lamport in 1981 [12]. In his scheme, the system would provide each remote user with an identity and a password that, when the user tended to login to the system, the identity was first inserted into the system for authentication and then the password was input. Not until the password was consistent with the one in the verification table would the user be allowed to log in to the system. However, later some security flaws, which made the whole authentication system insecure, were pointed out in the scheme [13], thus many improved schemes appeared in terms of security and practicability of authentication in the following years.

For instance, Hwang et al. proposed a single server authentication scheme [14]; Haller and Yeh applied S/KEY one-time password scheme [15]; and some researchers indicated the insecurity of S/KEY scheme [16].

Meanwhile, the Encrypted Key Exchange (EKE), as a family of password-authenticated key agreement methods, was proposed by Bellovin and Merritt [17]. The EKE could effectively amplify a shared password into a shared key, which might be further applied to a zero-knowledge password proof.

In 2000 and 2001, Sandirigama et al. and Lin et al. proposed SAS [18] and OSPA protocol [19], respectively, which were supposed to be superior to the Lamport's protocol, CINON protocol, and the PERM protocol [20], in terms of storage utilization, computing time, and transmission overhead. In 2002, Chen and Ku further proposed two malicious attacks on SAS and OSPA protocol [21].

Afterwards, with the convenience and popularity of smart cards, many smart card-based authentications and key agreement schemes were proposed to enhance the security and the efficiency of the systems [6, 22]. In those schemes, a client was only required to memorize a password and hold a personal smart card, in which a secret number issued by the server was stored. Also, it was not necessary for a remote server to pay extra cost for maintaining a security-sensitive verification table in those schemes.

Furthermore, Das et al. proposed a dynamic ID-based remote user authentication scheme based on password technology [23]. The scheme allowed the users to change and choose passwords freely without the server maintaining any password verifier table. It

was also secure against ID-theft attack, replay attack and other malicious attacks. However, Wang et al. indicated recently that it was completely insecure for its independence of the password [6].

In 2005, Fan et al. proposed a robust authentication scheme [24] based on the factoring problem. However, the key length of the server was long and the computation cost of the system was high in the scheme. Later, Wang et al. [7] further tried to solve these problems. Another schemes were also proposed later then. Liao et al proposed a single server authentication scheme; Song applied the concept of smart card to the password scheme [25]; and Tsai raised the method which was based on nonce for authentication protocol [26].

Recently, more authentication schemes are proposed in medical area. An ID-based system is published for long-distance medical care by Cao and Zhai [27]. Biometric scheme which applies Chaotic hash function is raised by Das et al. [28]. As well anonymity preserving scheme is proposed by Wen [29]. Further, dynamic-remote scheme is proposed by Huang and Wu [30].

Besides, cloud-based mutual authentication protocols have also been proposed for the use of medicine field, such as VPN agent system by Xie et al. [31], the anti-tracing scheme by Abughazalah et al. [32], the supply chain system from Lin et al. [33], and the enhanced scheme for forward security and anonymity protection proposed by Chen et al. [34].

In conclusion, for any proposed password-based user authentication schemes, the off-line password guessing attacks pose the biggest threat in terms of damage among various types of attacks [3, 5-6]. Endless possible problems are then presented after the hacking of the password, such as server spoofing attacks and impersonation attacks. In the most commonly seen trick of such attack, an adversary may masquerade as a legal user to access to the crucial information, intercept some transmitted values over the network, or generate various self-guessed parameters to hack the correct password of a specific user using brute force [3, 5-8]. Therefore, a new user authentication scheme with dynamic passwords is proposed in the following section.

3 The Proposed Scheme

In this section, we would like to present a user authentication scheme that incorporates the property of dynamic passwords into resisting on-line and off-line password guessing attacks. Besides, other common attacks such as replay attacks, stolen-verifier attacks, impersonate attacks, and server spoofing attacks will also be rendered ineffectual in our scheme.

This scheme is composed of three phases which include the registration phase, the login phase and the verification phase. The main entities include users and the remote server. All users begin the registration phase by obtaining their exclusive smart cards and the login passwords. Following, they can login to the remote server through smart cards to acquire the desired services after the server has verified their identities, passwords, and the transmitted parameters. Figure 1 is the flowchart of this proposal.

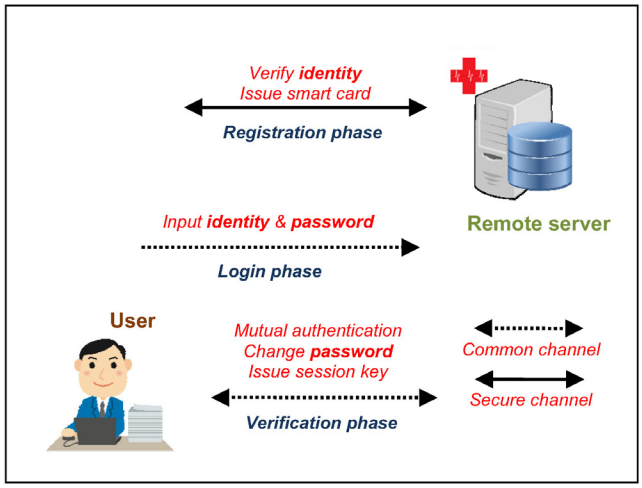


Figure 1. Flowchart of proposed scheme

Before describing the details of our proposal, the notation defined and used in this scheme is shown in Table 1.

Table 1. Notation defined and used in our scheme

U	the user
pw	the password of user U
pw_{new}	the new password of user U, changed after each authentication
ID	the identity of user U
S	the remote server
$h(\cdot)$	a public one-way hash function
$\oplus$	a bit-wise XOR operation
$\|\|$	a bit concatenation operator

3.1 Registration Phase

Suppose user U wants to register with a remote server S. He would first propose a registration request so as to get his password and his smart card from the server as follows.

Step 1: U submits his owned identity ID to S.

Step 2: S checks the validity of ID, and then computes the related hash value $v = h(K \oplus ID)$, where K is the fixed secret number belonging to S.

Step 3: S generates U's password $pw = [v]^i$, where $[v]^i$ means the random i bits of value v, $40 \le i \le 104$.

Step 4: S generates the value $s = h(pw \,||\, K)$. Note that s would be well protected by the device of smart card, and no other user can catch the value of s easily.

Step 5: S personalizes U's smart card by including the above parameters $[h(\cdot), i, s]$, and returns it with pw to U through a secure channel.

3.2 Login Phase

When user U wants to log into the remote server S, U firstly inserts his smart card into a terminal and then keys in his identification ID along with his password pw. The smart card will execute the following steps automatically:

Step 1: Read the stored values of the card s and i.

Step 2: Choose a random number N_c to compute C_1 and C_2, where $C_1 = h(N_c \| s)$, $C_2 = N_c \cdot pw$.

Step 3: Send out the user's ID, the values i, C_1, and C_2 together to the remote server S through the common network channel.

3.3 Verification Phase

When server S receives a login request (ID, i, C_1, C_2) from user U, server S does the verification as follows:

Step 1: Check the validity of user U's identity ID. If the ID is legal, S accepts the service request; otherwise, the service request is rejected.

Step 2: Restore the user U's password pw by using the secret value K: $v = h(K \oplus ID)$, and $pw = [v]^i$.

Step 3: Compute the unique value s of user through the hash function $h(pw \| K)$.

Step 4: Check whether the following equation holds true:

$$C_1 \stackrel{?}{=} h(\frac{C_2}{pw} \| s)$$

If the two values are the same, go to Step 5; otherwise, stop and reply the error message to U.

Step 1: Compute the new password pw_{new} by $[v]^{i'}$, where i' is another random i bits of value v.

Step 2: Use the pw_{new} to compute the new secret value $s' = h(pw_{new} \| K)$ and $C_3 = N_c \oplus pw_{new}$, $C_4 = h(s') \cdot i'$.

Step 3: Find an appropriate value N_s to make the sum of $s' \cdot pw_{new} + N_s$ being equal to the original secret value s.

Step 4: Send N_s, C_3, C_4 together with $C_5 = h(s \| s')$ to U through the common network channel.

3.4 Password Updating Phase

When user U receives the reply message (N_s, C_3, C_4, C_5) from the remote server S, U does the verification as follows:

Step 1: Obtain the new password pw_{new} by $C_3 \oplus N_c$.

Step 2: Compute the new secret s' through the equation $s' = (s - N_s) \cdot pw_{new}^{-1}$.

Step 3: Verify whether $h(s \| s')$ is equivalent to C_5. If they are equivalent, user U confirms that S is valid.

Step 4: Compute the i' through $C_4 \cdot h(s')^{-1}$.

Step 5: Send back $C_6 = h(s' \| s)$ to server S for another side authentication.

Step 6: Overwrite the value s stored in the smart card into s', and i into i'.

3.5 Data Transmission Phase

After the remote server S receives C_6:

Step 1: Compare C_6 with the value $h(s' \| s)$ calculated to check whether both of them are equivalent or not. If they are, U is authenticated and granted access and obtain services and resources of S. A session key $sk = h(s' \| h(s'))$ will be generated and used for secure transmission during the following operations after the mutual authentication process is done.

4 Security Analysis

A password-based user authentication scheme is secure when it can resist various malicious attacks, including replay attacks, stolen-verifier attacks, on-line and off-line password guessing attacks, server spoofing attacks, impersonation attacks, and denial of service attacks. Specifically, because the current schemes are ineffectual in fending off off-line password guessing attacks, schemes are imperfect and insecure, therefore, we propose a dynamic password-based user authentication scheme that can effectively resist such attacks. Following, we will analyze in detail and demonstrate how the scheme satisfies in resisting the above-mentioned attacks.

4.1 Replay Attacks

A replay attack is a kind of network attack where a valid data transmission is maliciously repeated again and again. This kind of attack may be carried out by machinated adversary, who intercepts the data and transmits it repeatedly. To prevent such attacks in our scheme, we make use of two fresh and random variables N_c and N_s during the authentication process. Suppose that an adversary intentionally intercepts (ID, i, C_1, C_2,) from the login phase, and seems to be capable of impersonating the legal user to log into the server by replaying the message. However, without random number N_c, he cannot take out the related parameters to compute the correct C_6 for the further confirmation of server S's identity, even though he receives the replied message (N_s, C_3, C_4, C_5) in the former verification phase. Furthermore, because the secret password changes with each round of authentication, if the attacker cannot effectively find out the correct values, his replied message will be rejected before it can execute the rest of the verification procedure. Therefore, the replay attacks will be fully failed.

4.2 On-line Password Guessing Attacks

On-line password guessing attacks occur when an attacker continuously guesses every possible password and tries to log into the server until he success. In our scheme, such attack will be recognized immediately.

Suppose an adversary attempts to validate the password of a legal user, he would guess a possible password to compute some parameters and start to execute the login phase. However, the probability of knowing the correct password is only 2^{-k}, where k is the length of the password. According to our scheme, the server can detect abnormality by confirming whether $h(C_2 \cdot pw^{-1} \| s)$ is equal to C_1. Generally, the third guess from the adversary is wrong therefore he will undoubtedly be kicked out of the system. Consequently, on-line password guessing attacks cannot work here.

4.3 Off-line Password Guessing Attacks

For any password-based user authentication scheme, the off-line password guessing attacks poses the biggest threat in terms of damage among the various types of attacks. In the most commonly seen method of such attack, an adversary will intercept some transmitted information or generate various self-guessed parameters to hack the correct password of a specific user by using brute force.

To render this kind of attack ineffectual, our scheme makes the password dynamic, i.e. the password used in each authentication procedure is different. Therefore, even if the adversary successfully hacks the password with brute force, the password will still be ineffectual, because the password will have changed, just as the password that will be used in the future for authentication will be different from the previous one. Furthermore, our scheme takes care in protecting s and s', the related parameters of new passwords. Anyone who attempts to get the two values must crack the cryptographic one-way hash function primarily, which is believed to be difficult to solve until now. Therefore, our scheme can withstand off-line password guessing attacks.

4.4 Stolen-Verifier Attacks

Stolen-verifier attacks mean that a machinated inside member can steal or modify passwords or verification tables of users stored in a server's database. In our scheme, since the user's password is recovered directly by a server by the way of using the parameters sent from the user at the verification phase, there is no need to keep passwords or verification tables in a server's database. Therefore, the inside member would not be able to steal or modify the passwords. This attack is meaningless here.

Moreover, the common problem of password synchronization in dynamic password-based user authentication scheme is also solved. Since no passwords or verification tables are stored in the server's database, the condition of inconsistent passwords, which are stored by the user and the server, would never happen.

4.5 Server Spoofing Attacks

Adversaries can masquerade the identity of the remote server, and then carry out illegal, imperceptible authentication behavior with other users so as to obtain the private information of user through the transmitted data. This is known as server spoofing attacks: someone masquerades as the server to cheat users.

Commonly, a conspiring attacker has two ways to successfully spoof users: one is through obtaining server's secret and impersonating as the server to authenticate with users, or by guessing the user's password and directly perform partial phases in the server part to communicate with users without the need of secret values. However, these methods are ineffective in our scheme. The secret value K is never transmitted through a common network channel therefore it is impossible for anyone to acquire. In addition, the user's password is hard to guess as it is protected by cryptographic hash function and random values. Therefore, the server spoofing attacks can be detected and will thus fail eventually.

4.6 Impersonation Attacks

Similar to server spoofing attacks, the impersonation attacks signify that someone masquerades as the other legitimate users to log into a server for acquiring services. Obviously, this situation will not arise in our scheme, since the password is protected with cryptographic hash function and random values. An adversary, in fact, cannot generate and interpret authentication messages correctly without the knowledge of a user's password. Consequently, a person who intends to masquerade as the user to acquire services is barred.

4.7 Denial of Service Attacks

The denial of service (DoS) attack restrains or inhibits communications between users and the remote server facilities. This attack may act on a specific user; for instance, an adversary may cause the server to reject the logins of a specific user until re-registration. The DoS attack rejects all or specific users by means of an offensive action on the server or by means of a falsification of the users' transmitted parameters. Then, the attacker can inconvenience the user but cannot imitate the user. In this proposal, the dynamic password allows a legal user to change the password each time so that the adversary cannot know the current verification information of the legal user. In addition, there is no password or verification table stored at the remote server, hence the scheme can prevent the adversary from using the effective methods to execute the DoS attack.

4.8 Smart Card Extraction Problems

When the smart card is lost or stolen, unauthorized users can easily extract the stored values of the smart

card and obtain some significant information such as personal identity, password, or login parameters by the physical extraction manner. This may cause the attacker to impersonate the user to login to the system and get some resources or services illegally. However, this scheme adopts dynamic passwords to prevent this situation that the different password would be used at each time when the user logins. Therefore, when attackers attempt to masquerade legal users to login by computing the transmitted value C_2 at login phase, they will fail without knowing the current password, even when they have got the users' smart cards and extracted the parameters [$h(\cdot)$, i, s]. Consequently, the proposed scheme can resist smart card extraction problems.

4.9 Perfect Forward Secrecy

When a user authentication scheme has perfect forward secrecy, it means an adversary cannot derive any previous used session keys to crack the encrypted documents, even though the user's password or the secret values are compromised by some malicious attacks. In our scheme, each session key is formed by the temporary value $h(s' \parallel h(s'))$. Whenever the communication ends between the user and the server, the session key will be revoked and no longer be used at the next round. When a user enters the system again, a new session key will be generated for him to encrypt the significant information during the current communication process. Therefore, it is very difficult for anyone to make use of all his known information so as to calculate any possible previous session key. We can declare that our scheme achieves perfect forward secrecy.

4.10 Man-in-the-Middle Attacks

Man-in-the-Middle attacks refer to attackers intercepting the data from the sender and the receiver, revising the contents, and re-sending the revised data to the sender and the receiver in the transmitting process so that they mistake that they are transmitting data to each other. The attack process is shown as follows. The sender A first transmits the required parameters to the receiver B. The middleman attacker C then intercepts the data and transmits the masqueraded parameter to the receiver B. B mistakes the data for the information delivered by A and then transmits the personal parameter back to A. C intercepts the data from B and further transmits fake data to A. Both A and B then generate the session key for encryption and decryption with the fake parameter; C therefore could transmit data with A or B but they do not realize that they are transmitting and exchanging data with C.

To prevent the attack from occurrence, this scheme would hide the random number N_c. Without random number N_c, an attacker cannot take out the related parameters to compute the correct C_6 for the further confirmation of server S to his identity, even though he receives the replied message (N_s, C_3, C_4, C_5) in the former verification phase. Furthermore, because the parameters change with each round of authentication, when the attacker cannot effectively identify the correct values, his replied message will be rejected before it can execute the rest of the verification procedure. Therefore, it could effectively avoid Man-in-the-Middle attacks.

4.11 Security Proof

Among all steps of the function, the most important protection of delivering parameter is a hash function. Hence, the utilizing hash function will be presented in this section as a secure method. Throughout the process, the hash function will be demonstrated as a "collision resistance" function which can establish a short and unique identifier.

Each transaction t_i is needed to be bound as the ledger state (t_1, …, t_{i-1}), so the ID of t_i can be conceived. Every ID is prior transactions and it is needed to be only n bits long. Hypothetically, this issue can be solved if one to one map H is existed from $\{0, 1\}^N$ to $\{0, 1\}^n$ for some large $N \gg n$ and then $H(t_1 \| \cdots \| t_i)$ would be the corresponding ID appending t_i to (t_1, …, t_{i-1}). However, it cannot work properly because $-2N$ is much bigger than $2n$ and there is no one to one map from a large set to a smaller set. Therefore, this problem is solved by utilizing the function H which is necessarily and essentially to be one to one.

A collision-resistant hash function is a map that covers from a large universe to a small one that is "practically one to one." In the sense that collisions for the function do exist while they are hard to find.

The following result is the main idea which can be thought of the "birthday paradox."

Lemma. If H is a random function from some domain S to $\{0, 1\}^n$, then T queries an attacker finds $x \neq x'$ such that $H(x) = H(x')$ is at most $T^2/2n$.

Proof. Let us think of H in the "lazy evaluation" mode. A random answer is chosen in $\{0, 1\}^n$ when the same time it is made. (We can assume the adversary never makes the same query twice since a repeat query can be simulated by repeating the same answer.) For $i < j$ in [T], let $E_{i,j}$ be the event that $H(x_i) = H(x_j)$. Since $H(x_j)$ is chosen randomly and independently from the prior choice of $H(x_i)$, the probability of $E_{i,j}$ should be 2^{-n}. Thus the probability of the union of $E_{i,j}$ covers all i, j's is less than $T^2/2^n$, but this probability is exactly what we need to calculate.

This means that a random function H is collision resistant in the sense that it is hard for an efficient adversary to find two inputs that collide. Thus the random oracle heuristic would suggest that a cryptographic hash function can be used to obtain the following object.

Definition (hash functions with collision resistant).

A collection $\{h_k\}$ of functions where h_k:{0, 1}*→{0, 1}^{n} for $k\in\{0, 1\}^n$ is a collision resistant hash function collection. If the map $(k, x) \mapsto h_k (x)$ is efficiently computable and for every efficient adversary A, the probability over k that $A(k)=(x, x')$ such that $x\neq x'$ and $h_k(x)= h_k(x')$ is negligible.

4.12 Comparison Table

To show how our proposed dynamic password-based user authentication scheme is suitable and efficient to be implemented, the comparison of our scheme with other related schemes are presentedas summarized in Table 2. Clearly, Wang et al.'s [6], Awasthi-Srivastava's [35], and Hwang-Wu's [9] schemes were all suffered from insecure attacks. The scheme proposed by Wang et al. was insufficient against off-line password guessing attacks and smart card extraction problems. Awasthi and Srivastava proposed the user authentication method based on symmetric cryptosystem and suitable in tele-medicine field; however, off-line password guessing attacks, denial of service attacks, and smart card extraction problems could not be withstood. Hwang and Wu allowed the users to dynamically change their personal passwords, yet smart card extraction problems were still happened. On the contrary, our scheme only requires few hashing functions and multiplication computations. With the analysis of the nine security concerns mentioned above, security on using the mechanism is assured.

Table 2. Comparison table of computation costs and security attacks

	Wang et al. (2009)	Awasthi and Srivastava (2013)	Huang and Wu (2017)	Our Proposal (2019)
Computation cost in registration phase	2H	2P+3H	2H	1H+1M
Computation cost in login phase	2H	3H	2H	1H+1M
Computation cost in verification phase	5H	4H	7H	11H+4M
Replay attacks	O	O	O	O
On-line password guessing attacks	O	O	O	O
Off-line password guessing attacks	X	X	O	O
Stolen-verifier attacks	O	O	O	O
Server spoofing attacks	O	O	O	O
Impersonation attacks	O	O	O	O
Denial of service attacks	O	X	O	O
Man-in-the-Middle attacks	O	O	O	O
Smart card extraction problems	X	X	X	O

H: one way hash function operations;
P: public key encryption/decryption operations;
M: multiplication operations;
O: achieve the prevention of malicious attacks;
X: cannot achieve the prevention of malicious attacks.

4.13 Time Simulation Analysis

Five measurements were made for each case of an average calculated value. Time is showed by milliseconds per 10^6 calculations. The system was run on a 64 bits Windows 10 with 1 core Intel i7 2.60GHz powered by 16GB RAM.

Choosing a suitable one-way hash function is needed to implement the authentication scheme. To find a better execution time of hash function, the following experiments should be done.

Exp 1: sixty-six characters are randomly chosen from the data that need to be encoded. Time cost of four different hash function method are shown in Table 3.

Table 3. Time cost of different hash function methods

Hash	1st (ms)	2nd (ms)	3rd (ms)	4th (ms)	Average (ms)
MD5	647	628	625	634	633.5
SHA-1	618	589	620	603	607.5
SHA-256	736	727	734	717	728.5
SHA-512	1065	1075	1040	1042	1055.5

Exp 2: fifty characters are randomly chosen from the encoded data. Time cost of four different hash function method are shown in Table 4.

Table 4. Time cost of different hash function methods

Hash	1st (ms)	2nd (ms)	3rd (ms)	4th (ms)	Average (ms)
MD5	757	799	808	747	777.75
SHA-1	778	775	783	764	775
SHA-256	852	887	839	862	860
SHA-512	1171	1163	1169	1175	1169.5

Exp 3: seventy-two characters are randomly chosen from the data that need to be encoded. Time cost of four different hash function method are shown in Table 5.

Table 5. Time cost of different hash function methods

Hash	1st (ms)	2nd (ms)	3rd (ms)	4th (ms)	Average (ms)
MD5	829	866	817	836	837
SHA-1	886	960	914	885	911.25
SHA-256	1127	1231	1161	1157	1169
SHA-512	1131	1126	1122	1100	1119.75

Exp 4: eighty-five characters are randomly chosen from the encoded data. Time cost of four different hash function method are shown in Table 6.

Table 6. Time cost of different hash function methods

Hash	1st (ms)	2nd (ms)	3rd (ms)	4th (ms)	Average (ms)
MD5	1049	1025	1012	1027	1028.25
SHA-1	1003	1013	1017	996	1007.25
SHA-256	1257	1241	1255	1243	1249
SHA-512	1238	1227	1236	1223	1231

Experiment results. the tables give the use comparison of MD5, SHA-1, SHA-256 and SHA-512 cryptographic hash functions. As MD5 and SHA-1 are endanger and not secure, they need to be avoided even though they are several times faster than the secure SHA-2 ones. When choosing cryptographic hash function, everything depends on the context of usage. Benchmark tests for this context is needed.

Besides, the use of multiplication is another operation of proposed scheme. We analysis the execution time of Karatsuba algorithm that is excellent method for performing multiplicative operations and investigating the multiplicand and multiplier having 4, 8, 16 and 32 bit length. Moreover, the performance of Karatsuba algorithm is analyzed in terms of the number of multiplication and the total of processing time. The applications used for performance analysis are implemented using MATLAB R2014a and the computer used for testing has these features: Windows 7 64 bit Operating System, Intel Core i5-3317U CPU 1.70 GHz Processor and 4 GB RAM. The performance analysis of Karatsuba algorithm in terms of the number of multiplication for different bit lengths is given in Figure 2.

The bit length increases along with the number of multiplication due to the processing of Karatsuba algorithm. In addition, the more the number of multiplication raises, the more amount of hardware increases. Therefore, the cost required to perform the multiplication operation rises. When compared to each other, the number of multiplication of Karatsuba algorithm is less than classical multiplication method. The performance of Karatsuba algorithm in terms of the total process time for different bit lengths is analyzed as shown in Figure 3.

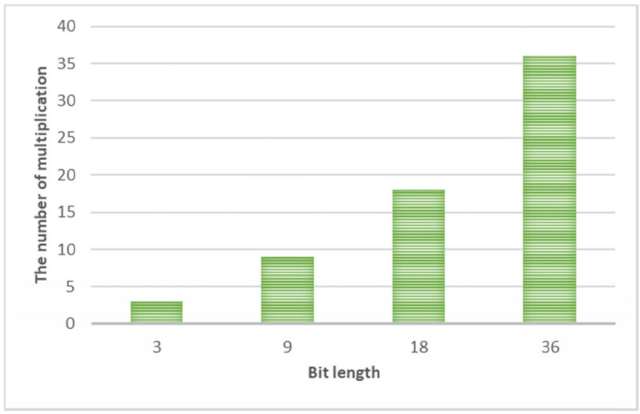


Figure 2. Performance analysis with the number of multiplication number

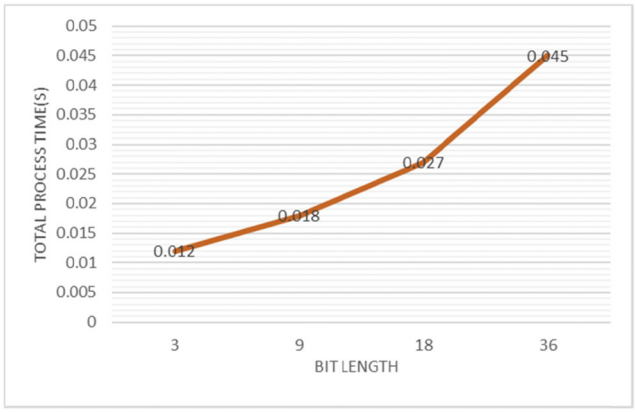


Figure 3. Performance analysis with the total process time

5 Conclusions

In this paper, a dynamic password-based user authentication scheme is proposed to secure an information exchange of PHRs from offline password guessing attacks. The passwords' changing ability will make the attacker difficult to catch the correct password, which means it gives a more protection for user from the attacks. Furthermore, it can also prevent another type of attacks that may occur after obtaining the password such replay attacks, server spoofing attacks and impersonation attacks. The comparison of the experiment results showed that our scheme has achieved a perfect forward secrecy that makes attackers difficult to crack and modify any previous encrypted documents. Therefore, it proves that our scheme has more strength, secure and efficient to be implemented.

Acknowledgements

The paper is funded by the Ministry of Education in Taiwan for developing the characteristics of national vocational and technological colleges and universities,

with the Leap Program Number 108G0038 of the 108th academic year.

References

[1] T. Heart, O. Ben-Assuli, I. Shabtai, A Review of PHR, EMR and EHR Integration: A More Personalized Healthcare and Public Health Policy, *Health Policy and Technology*, Vol. 6, No. 1, pp. 20-25, March, 2017.

[2] J.-C. Liu, C.-H. Lin, K.-Y. Lee, Cloud-based Personal Data Protection System and Its Performance Evaluation, *Journal of Internet Technology*, Vol. 20, No. 6, pp. 1721-1727, November, 2019.

[3] Z. Y. Wu, Y. F. Chung, F. Lai, T. S. Chen, A Password-based User Authentication Scheme for the Integrated EPR Information System, *Journal of Medical Systems*, Vol. 36, No. 2, pp. 631-638, April, 2012.

[4] C. H. Lin, Y. Y. Lai, A Flexible Biometrics Remote User Authentication Scheme, *Computer Standards & Interfaces*, Vol. 27, No. 1, pp. 19-23, November, 2004.

[5] R. Lu, Z. Cao, Z. Chai, X. Liang, A Simple User Authentication Scheme for Grid Computing, *International Journal of Network Security*, Vol. 7, No. 2, pp. 202-206, September, 2008.

[6] Y. Y. Wang, J. Y. Liu, F. X. Xiao, J. Dan, A More Efficient and Secure Dynamic ID-based Remote User Authentication Scheme, *Computer Communications*, Vol. 32, No. 4, pp. 583-585, March, 2009.

[7] X. M. Wang, W. F. Zhang, J. S. Zhang, M. K. Khan, Cryptanalysis and Improvement on Two Efficient Remote User Authentication Scheme Using Smart Cards, *Computer Standards & Interfaces*, Vol. 29, No. 5, pp. 507-512, July, 2007.

[8] E. J. Yoon, E. K. Ryu, K. Y. Yoo, Further Improvement of an Efficient Password Based Remote User Authentication Scheme Using Smart Cards, *IEEE Transactions on Consumer Electronics*, Vol. 50, No. 2, pp. 612-614, May, 2004.

[9] T. C. Yeh, H. Y. Shen, J. J. Hwang, A Secure One-time Password Authentication Scheme Using Smart Cards, *IEICE Transactions on Communications*, Vol. E85-B, No. 11, pp. 2515-2518, November, 2002.

[10] C.-Y. Chen, T.-C. Hsu, H.-T. Wu, J. Y. Chiang, W.-S. Hsieh, Anonymous Authentication and Key-Agreement Schemes in Vehicular Ad-Hoc Networks, *Journal of Internet Technology*, Vol. 15 No. 6, pp. 893-902, November, 2014.

[11] S.-K. Kim, M. G. Chung, More Secure Remote User Authentication Scheme, *Computer Communications*, Vol. 32, No. 6, pp. 1018-1021, April, 2009.

[12] L. Lamport, Password Authentication with Insecure Communication, *Communications of the ACM*, Vol. 24, No. 11, pp. 770-772, November, 1981.

[13] R. Song, L. Korba, and G. Yee, Analysis of Smart Card-based Remote User Authentication Schemes, *Proceedings of the 2007 International Conference on Security and Management*, Las Vegas, Nevada, USA, 2007, pp. 323-329.

[14] T. Hwang, Y. Chen, and C. S. Laih, Non-interactive Password Authentications Without Password Tables, *IEEE Region 10 Conference on Computer and Communication systems*, Hong Kong, 1990, pp. 429-431.

[15] N. Haller, The S/KEY (TM) One-time Password System, *Proceedings of Internet Society Symposium on Network and Distributed System Security*, San Diego, California, USA, 1994, pp. 151-158.

[16] C. J. Mitchell, L. Chen, Comments on the S/KEY user Authentication Scheme, *ACM Operating Systems Review*, Vol. 30, No. 4, pp. 12-16, October, 1996.

[17] S. M. Bellovin, M. Merritt, Encrypted Key Exchange: Password-Based Protocols Secure Against Dictionary Attacks, *Proceedings of the IEEE Symposium on Research in Security and Privacy*, Oakland, CA, USA, 1992, pp. 72-84.

[18] M. Sandirigama, A. Shimizu, M. T. Noda, Simple and Secure Password Authentication Protocol (SAS), *IEICE Transactions on Communications*, Vol. E83-B, No. 6, pp. 1363-1365, June, 2000.

[19] C. L. Lin, H. M. Sun, T. Hwang, Attacks and Solutions on Strong-password Authentication, *IEICE Transactions on Communications*, Vol. E84-B, No. 9, pp. 2622-2627, September, 2001.

[20] A. Shimizu, T. Horioka, H. Inagaki, A Password Authentication Method for Contents Communications on the Internet, *IEICE Transactions on Communications*, Vol. E81-B, No. 8, pp. 1666-1763, August, 1998.

[21] C.-M. Chen, W.-C. Ku, Stolen-verifier Attack on Two New Strong-password Authentication Protocols, *IEICE Transactions on Communications*, Vol. E85-B, No. 11, pp. 2519-2521, November, 2002.

[22] W.-S. Juang, S.-T. Chen, H.-T. Liaw, Robust and Efficient Password-authenticated Key Agreement Using Smart Cards, *IEEE Transactions on Industrial Electronics*, Vol. 55, No. 6, pp. 2551-2556, June, 2008.

[23] M. L. Das, A. Saxena, V. P. Gulati, A Dynamic ID-based Remote User Authentication Scheme, *IEEE Transactions on Consumer Electronics*, Vol. 50, No. 2, pp. 629-631, May, 2004.

[24] C.-I. Fan, Y.-C. Chan, Z.-K. Zhang, Robust Remote Authentication Scheme with Smart Cards, *Computers & Security*, Vol. 24, No. 8, pp. 619-628, November, 2005.

[25] R. Song, Advanced Smart Card Based Password Authentication Protocol, *Computer Standards & Interfaces*, Vol. 32, No. 5-6, pp. 321-325, October, 2010.

[26] J. L. Tsai, Efficient Nonce-based Authentication Scheme for Session Initiation Protocol, *International Journal of Network Security*, Vol. 9, No. 1, pp. 12-16, July, 2009.

[27] T. Cao, J. Zhai, Improved Dynamic Id-based Authentication Scheme for Telecare Medical Information Systems, *Journal of Medical Systems*, Vol. 37, No. 2, Article number 9912, April, 2013.

[28] A. K. Das, A. Goswami, An Enhanced Biometric Authentication Scheme for Telecare Medicine Information Systems with Nonce Using Chaotic Hash Function, *Journal of Medical Systems*, Vol. 38, No. 6, Article number 27, June, 2014.

[29] F. Wen, A Robust Uniqueness-and-Anonymity-Preserving Remote User Authentication Scheme for Connected Health Care, *Journal of Medical Systems*, Vol. 37, No. 6, Article number 9980, December, 2013.

[30] K. K. Huang, Z. Y. Wu, A Reliably Dynamic User-remote Password Authentication Scheme for Medical Environments, *Journal of Internet Technology*, Vol. 18, No. 5, pp. 995-1001, September, 2017.

[31] W. Xie, L. Xie, C. Zhang, Q. Zhang, C. Tang, Cloud-based RFID Authentication, *2013 IEEE International Conference on RFID*, Penang, Malaysia, 2013, pp. 168-175.

[32] S. Abughazalah, K. Markantonakis, K. Mayes, Secure Improved Cloud-based RFID Authentication Protocol, in: J. Garcia-Alfaro, J. Herrera-Joancomarti, E. Lupu, J. Posegga, A. Aldini, F. Martineli, N. Suri (Eds.), *Data Privacy Management, Autonomous Spontaneous Security, and Security Assurance*, LNCS, Vol. 8872, Springer, 2015, pp. 147-164,

[33] I. C. Lin, H. H. Hsu, C. Y. Cheng, A Cloud-based Authentication Protocol for RFID Supply Chain Systems, *Journal of Network & Systems Management*, Vol. 23, No. 4, pp. 978-997, October, 2015.

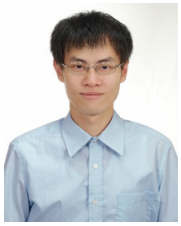
[34] M. M. Chen, Q. K. Dong, L. L. Li, Cloud-based RFID Mutual Authentication Protocol, *Journal of Cryptologic Research*, Vol. 5, No. 3, pp. 231-241, June, 2018.

[35] A. K. Awasthi, K. Srivastava, A Biometric Authentication Scheme for Telecare Medicine Information Systems with Nonce, *Journal of Medical Systems*, Vol. 37, No. 5, Article number 9964, October, 2013.

Biographies

Wun-Lin Chen received the M.D. degree in National Defense Medical Center of Taiwan in 2007. He is currently attending physician in the Division of Traumatology, the Department of Surgery at Tri-Service General Hospital, Taiwan. His current interests focus on medical information, electronic medical record, and teleconsultation.

Tias Kurniati received a Master's degree in Information Management from the National Taiwan University of Science and Technology in 2018. She is currently a Ph.D. student at Tunghai University focusing on the information management field under the Department of Statistics. Her main areas of research interest are deep learning, image and text recognition, computer graphics and information security.



Zhen-Yu Wu received the Ph.D. degree in Computer Science from National Taiwan University in 2011. He is currently an Associate Professor with the Department of Information Management at National Penghu University of Science and Technology, Taiwan. His current interests focus on information security, cryptography, Internet of Things, and medical information.

Yu-Min Huang received her Ph.D. in Statistics from University of Minnesota-Twin Cities, US. Currently, she is working as assistant professor in the Department of Statistics at Tunghai University. Her research interests include time series, multivariate data analysis, network data analysis, quantitative modeling, high dimensional data.

Sheng-Der Hsu received the Ph.D. degree in Education Science from National Taiwan Normal University in 2018. He is currently director in the Division of Traumatology, the Department of Surgery at Tri-Service General Hospital, Taiwan. His current interests focus on medical information, electronic medical record, and teleconsultation