J Med Syst (2012) 36:2547–2555
DOI 10.1007/s10916-011-9727-9

ORIGINAL PAPER

A Reliable User Authentication and Key Agreement Scheme for Web-Based Hospital-Acquired Infection Surveillance Information System

Zhen-Yu Wu · Yi-Ju Tseng · Yufang Chung · Yee-Chun Chen · Feipei Lai

Received: 25 February 2011 / Accepted: 25 April 2011 / Published online: 10 May 2011

Abstract With the rapid development of the Internet, both digitization and electronic orientation are required on various applications in the daily life. For hospital-acquired infection control, a Web-based Hospital-acquired Infection Surveillance System was implemented. Clinical data from different hospitals and systems were collected and analyzed. The hospital-acquired infection screening rules in this system utilized this information to detect different patterns of defined hospital-acquired infection. Moreover, these data were integrated into the user interface of a signal entry point to assist physicians and healthcare providers in making decisions. Based on Service-Oriented Architecture, web-service techniques which were suitable for integrating heterogeneous platforms, protocols, and applications, were used. In summary, this system simplifies the workflow of hospital infection control and improves the healthcare quality. However, it is probable for attackers to intercept the process of data transmission or access to the user interface. To tackle the illegal access and to prevent the information from being stolen during transmission over the insecure Internet, a password-based user authentication scheme is proposed for information integrity.

Keywords Hospital-acquired infection · Surveillance system · Infection control · Illegal access · Authentication

Z.-Y. Wu (✉) · F. Lai
Department of Computer Science and Information Engineering,
National Taiwan University,
Taipei, Taiwan
e-mail: d96922021@ntu.edu.tw

Y.-J. Tseng · F. Lai
Graduate Institute of Biomedical Electronics and Bioinformatics,
National Taiwan University,
Taipei, Taiwan

F. Lai
Department of Electrical Engineering,
National Taiwan University,
Taipei, Taiwan

Y. Chung
Department of Electrical Engineering,
Tunghai University,
Taipei, Taiwan

Y.-C. Chen
Center for Infection Control,
National Taiwan University Hospital,
Taipei, Taiwan

Y.-C. Chen
Department of Internal Medicine, National Taiwan University
Hospital and College of Medicine,
Taipei, Taiwan

Introduction

Hospital-acquired infection (HAI) is a disease developed with the admission into the hospital as well as a treatment consequence performed by the medical staff. Normally, a disease is considered a hospital-acquired infection when it has developed 72 h after the admission to the hospital. Nosocomial or hospital-acquired infections are the major threats to the safety of all in-patients and one of the most important problems in hospital. According to the past research, HAI would raise healthcare costs, prolong length of hospital stay, and increase mortality [1–4].

In recent years, considerable concern has been arisen for the computer-assisted HAI surveillance system in infection control research. Several studies have suggested the benefits of HAI surveillance, such as reducing the incidence of HAI and improving the quality of healthcare. Numerous

benefits were conducted in the US [5] 30 years ago and validated in Germany [6] recently. As mentioned above, surveillance of hospital-acquired infections is important and indispensable. In order to improve patient safety, a web-based HAI surveillance information system (WHISS) is designed as a service, which not only handles the complex electronic health records (EHR) consistently and efficiently, but also provides stable and consistent results of HAI screening. In addition, for transferability and scalability, this system follows the service-oriented architecture (SOA) and communicates with Health Level Seven (HL7) message.

The process of WHISS is described as follows. First, the required data for the WHISS screening are transmitted from different hospitals. Then, the HAI candidates are automatically detected according to the screening rules established by domain experts. Finally, these system-detected HAI cases are displayed in a web-based user-friendly interface based on XML via browsers. The interface also provides all helpful information, such as operation information and order information, allowing infection control nurses to get all the necessary data to reconfirm the HAI cases.

Obviously, the private data and information in the WHISS are transmitted over insecure networks. How to provide a secure environment for data transmission and communication between users and the WHISS becomes a significant issue. In order to prevent illegal users from visiting the remote server, secure strategies over an insecure network are concerned. Several relevant user authentication schemes and secret-key distribution protocols have been proposed [7–11], in which the password-based authentication mechanism is employed mostly because of its efficiency [12–14]. Under such mechanism, each user is allowed to select personal password and keep in mind without any additional assistant device for further authentication processes.

Although the password is endowed with advantageous properties of simplicity and human memory, it can easily succumb to attacks with brute force, such as off-line guessing attacks from which various existing schemes are suffering, or to spoofing and impersonation problems once the password is hacked.

Therefore, an ameliorative password-based authentication scheme is proposed in this paper, achieving to resist off-line password guessing attacks, replay attacks, on-line password guessing attacks, and other attacks. In light of security, the proposed scheme is provided with good practicability for the web-based hospital-acquired surveillance information system.

The rest of this paper is organized as follows. "System architecture" introduces the web-based HAI surveillance information system architecture. "Related technologies" illustrates the proposed password-based user authentication scheme. Security analyses are completed in "Proposed scheme". Comparisons are given in "Security analysis"; and finally, conclusions are drawn in "Comparison".

System architecture

The architecture of the WHISS is displayed in Fig. 1. There are four major parts in the system, including authentication and communication service, related data collection service, HAI screening service, and HAI screening result analysis and modification service.

Authentication and communication service

This service validates users' authentications via SOA mechanism and provides communication and connectivity via HL7 framework. The SOA is a flexible set of design

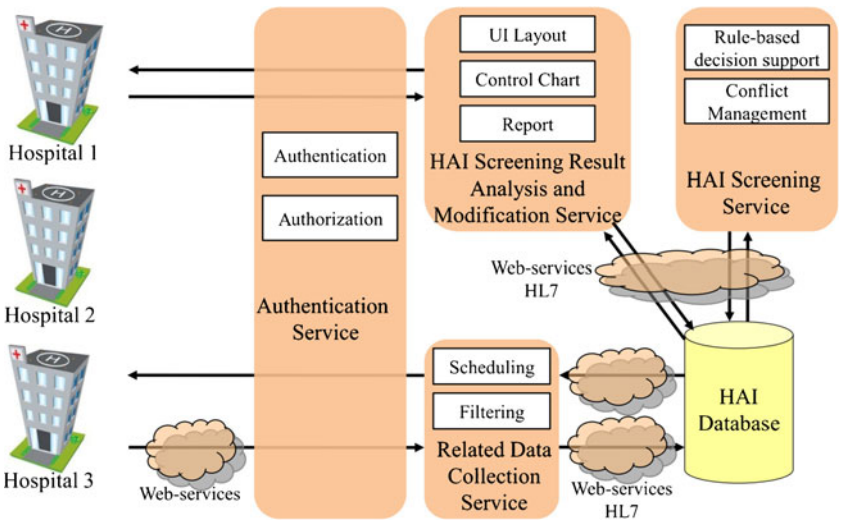

Fig. 1 System architecture of WHISS

principles used during the phases of systems development and the integration in computing, representing the current pinnacle of interoperability, in which resources on a network are available as individual, loosely-coupled, and independent services [15–18]. The HL7 framework is the middleware integration engine of the WHISS architecture. It supports message management, routing, mapping, and database access. The HL7 middleware accesses to HL7 messages, embedded in Extensible Markup Language (XML) format, over the Simple Object Access Protocol (SOAP) [19, 20]. In order to achieve the data consistency, a data exchange server that only receives the message sending from the HL7 middleware, is introduced. While the data exchange server receiving messages, it will perform the data synchronization among WHISS [21].

The WHISS system is accessible, by all authorized screening program professionals, to the hospital-enabled unique identification of patients, the related data, and the screening results. In addition, the medical staff can conduct mathematical and statistical analyses on the screening results in the database. During the screening procedures, all operations of database transactions and data exchanges among systems are based on the HL7 middleware framework standards.

Related data collection service

For hospital-acquired infection screening, the related data should be collected, such as demographic data, diagnoses, medication, vital signs, orders, and laboratory reports (Table 1). To be screened for HAI, each hospital transmits the related data to the HAI screening service through the web-services with the secure user authentication and the key agreement scheme in this research. These data are kept in the database after filtering and formatting, and all information required by the HAI screening service is collected via HL7 message [22].

HAI screening service

The screening rules of hospital-acquired infection surveillance were established by infection control specialists and computer engineers based on literature and local epidemiology [23]. The system provided automatic screening services of urinary tract infection (UTI) and blood-stream infection (BSI). For example, there are two major types of hospital-acquired UTI, namely Type 1 symptomatic UTI (SUTI) and Type 2 asymptomatic UTI (ASB). It must be noted that patients might match different types' definitions at the same time. The algorithm to avoid duplicate case detection is based on the priority of HAI types determined by infection control nurses. For example, if a candidate is marked as Type 1, the process of candidate detection of this case would be stopped and never be checked any other types that the priority lower than Type 1. It means that each HAI case only belongs to one type of HAI. In the HAI screening service, the data which collected by related data collection services are used to distinguish infection and non-infection. The basic ideas of HAI screening are having positive culture result and occurring symptom at the same time.

Apart from this, not all candidates should be established as HAI cases. After screening, candidates will be eliminated if there are conflicts among candidates and existed HAI events which was already in the infection cases database. This process follows merged guidelines provided by infection control nurses. The basic idea of conflict merging is that the period of two HAI cases of the same patient must be greater than and equal to specific days. With the example from hospital-acquired blood-stream infection, the highest frequency for establishing a case is 2 days. Abiding by this idea, the new microorganism, which is detected as HAI, should be added into the former event of this patient when the gap between the date of new microorganism and existed candidate is shorter than 7 days. Moreover, many other merging guidelines are also involved. In another situation, if the microbiology

Table 1 Related data list in the data collection service

Data	Contents
Demographic data	person ID, admission date, discharge date, birthday
Order data	order code, order type, order establish date
Lab order data	order code, order type, order establish date, specimen information
Lab result data	specimen information, laboratory result
Diagnosis data	diagnosis code, diagnosis name, diagnosis date
Symptom data	symptom name, symptom value, symptom onset date
Hospital-acquired Infection Case Data	infection type, infection date, infection reason, microorganism information of infection
Operation data	operation name, operation date, operation surgeon ID
Pharmacy data	drug name, dose, start date, use day, frequency

report of the candidate displays the same as any other HAI events, and there is no evidence to prove that the patient has recovered, the candidate will also be rejected. The evidence of recovery includes the disappearance of associated symptoms and negative results of the micro-organism culture test.

To evaluate the screening rules, the hospital-acquired UTI data between June 2009 and December 2009 were evaluated in previous research [24]. The accuracy of this system was 81.9%, and the square of the sample correlation coefficient of HAI events count with and without the WHISS in different departments was 0.982. In conclusion, the HAI screening service provides most candidates of HAI events; and, the tendency and the distribution of HAI events with screening rules are similar to manually collected HAI events.

HAI screening result analysis and modification service

The final step is importing the HAI cases into the HAI screening result analysis and the modification service in WHISS, which offers a single entry point through web-based interface via browser. HAI cases and the related data are displayed on the interface based on XML. The information of infection cards is described by XML. The tag name stands for the field name of data, such as infection date and infection type, and the element value stands for the content in the data filed. The information is showed by the interface comprising basic data of each person, such as diagnoses, operation, medication, microbiology reports of culture results and antibiotic resistance tests. There is a lack of integrated computerized medical services that can assist physicians and healthcare providers in differentiating HAIs in all infected patients. The provision of these integrated infection-related data could advance the accuracy and the speed of reconfirmation of HAI cases [25]. Moreover, instead of entering data manually, infection control nurses could search and modify these automatically loaded events by web-based HAI screening results analysis and modification service quickly and longitudinally, then output data sheet they need as a report. The service also provides some statistical functions, such as the tendency of quantity of infection cases in a specific department, which is of great interest to infection control specialists [26]. Furthermore, for special cases, the system provides infection control nurses with a HAI cases input and modifies the interface, allowing nurses to add and edit the information of HAI cases.

Related technologies

The related cryptographic and mathematical methods are applied in the proposed authentication scheme. They are hash function and symmetric cryptography, and are explained as follows.

Hash function

A hash function is a mathematical function that converts a variable-size message into a small fixed-size digest [31]. Assume that a hash value H is generated by a hash function h, it can be denoted by $H=h(M)$, where M is a message of variable length and the hash value H is of a fixed length. One normally used cryptographic hash function is the secure hash algorithm SHA-256 whose fixed-size output digest has a length of 256 bits. Besides, SHA-384 or SHA 512 algorithm will be able to replace the recommended SHA-256 algorithm if there are security or efficiency concerns in the future.

The function h should contain the following properties.

1. The output of h is limited to a fixed length in spite of the length of the input.
2. $h(x)$ is efficient to derive any given x so that the implementation of both hardware and software can be more practicable.
3. For any given x, it is easy to compute $h(x)=y$. On the contrary, it is hard to compute $h^{-1}(y)=x$ when y is given.
4. For any given block x, it is computationally infeasible to find $y\neq x$ but satisfy $h(y)=h(x)$. This is defined as weak collision resistance.
5. It is computationally infeasible to find any pair (x, y) so that $h(x)=h(y)$. This is defined as strong collision resistance.

Symmetric cryptography

Cryptography is used to transform a plaintext into an unreadable ciphertext to prevent unauthorized disclosure. It can mainly be classified into two different groups, as symmetric and asymmetric cryptographies [31].

In the manner of symmetric cryptography, both interaction parties, a client and a server, would share a key called session key. While a communication is executed, the sender uses the key to encrypt messages transmitted over the Internet. By using the same session key, the receiver can decrypt the messages. Therefore, the session key plays an important role to protect the confidential data from being revealed, modified, or deleted during transmission. It is the kernel of symmetric cryptography.

Advanced Encryption Standard (AES) [31], one of the most famous symmetric cryptographies nowadays, has vastly superior security and good throughput, so that it is recommended in many electronic applications, including e-medicine, for its proper session key to guarantee confidentiality and efficiency.

Proposed scheme

In this section, a user authentication scheme is proposed that it is not only suitable for the WHISS as "Authentication and communication service" described, but is also secure against various malicious attacks. This scheme is composed of three phases. They are the registration phase, the authentication phase, and the password change phase. Below is the detailed description of the proposal.

Before describing the details of the proposal, the notation defined and used in this scheme is shown in Table 2.

Registration phase

This phase is brought out whenever user U initially registers or re-registers to a remote server S. The U proposes a registration request so as to get his password and his smart card from the server as follows (Fig. 2).

Step 1 U selects a random number b and computes the hash value $h(b \oplus pw)$.

Step 2 U sends his ID, $h(pw)$, and $h(b \oplus pw)$ to S through a secure channel.

Step 3 S computes $P = h(ID \oplus x)/h_1(PI)$ and $R = P \oplus h(b \oplus pw)$, where PI is a set of messages. The messages will contain some random personal information such as names of pets, birthdays of families, and cell phone numbers of friends. The set needs to be guaranteed that is quite confused so as to be difficult to be guessed from other persons.

Step 4 S personalizes U's smart card included with the parameters $[h(.), h_1(.), R]$ and returns the card to U through a secure channel.

Step 5 U enters the generated value b into the smart card. The U's smart card finally contains the values b, R, $h_1(.)$, and $h(.)$.

Authentication phase

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

Step 1 Perform the following computations:

$C_1 = h_1(PI).(R \oplus h(b \oplus pw))$.

$C_2 = h(C_1 \oplus T_U)$, where T_U is the current time-stamp of U's computer.

$C_3 = r_1.C_1$, where r_1 is a random value.

Step 2 Send $C = \{ID, T_U, C_2, C_3\}$ to S through the common channel.

When server S receives a login request C at time T_S, the current timestamp of S's computer, server S does the verification as follows:

Step 1 Check the validity of ID, T_U, and the time interval T_S - T_U. If T_S - $T_U \leq 0$ holds, S will accept the login request; otherwise, the login request is rejected.

Step 2 Compute the hash value $h(h(ID \oplus x) \oplus T_U)$ and compare with the received C_2 to see whether the two values are equal or not. If it is, the login request is accepted; otherwise, the login request is rejected.

Step 3 Compute $r_1 = C_3/h(ID \oplus x)$.

Step 4 Compute $C_4 = h(h(ID \oplus x) \oplus h(T_S))$.

Step 5 Send C_4, and T_S to user U for a mutual authentication processing.

When user U receives the reply message (C_4, T_S), he does the verification as follows:

Step 1 Check the validity of T_S, and the time interval T_S - T_U. If T_S - $T_U = 0$ holds, U will terminate this session.

Step 2 Compute the hash value $h(C_1 \oplus h(T_S))$ and verify whether C_4 is equivalent to this value. If they are equivalent, U confirms that S is valid.

Step 3 Compute a session key $sk = h(r_1)$.

All above-mentioned steps are shown in Fig. 3.

Password change phase

When user U wants to change his password, he inserts his smart card into a terminal device. He firstly keys in his ID and his old password pw and then follows his new

Table 2 Notation defined and used in our scheme

U	the user
pw	the password of user U
ID	the identity of user U
S	the remote server
$h(.)$, $h_1(.)$	public one-way hash functions
$\oplus$	a bit-wise XOR operation

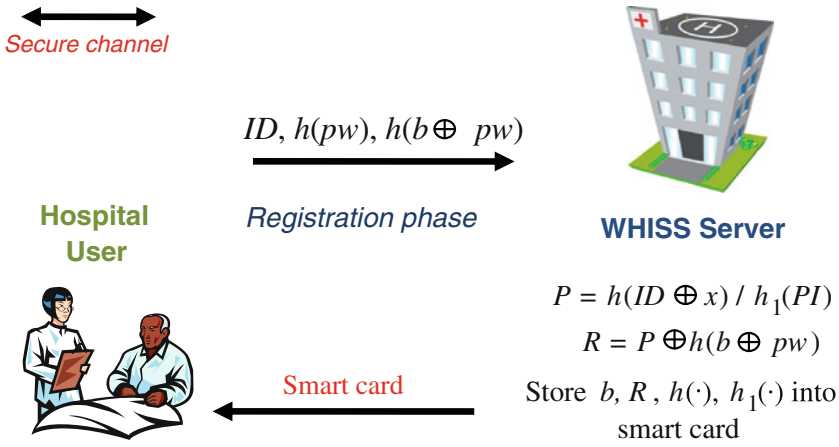

Fig. 2 Registration phase

password pw_{new}. The smart card will execute the following steps:

Step 1: Compute $R_{new} = R \oplus h(b \oplus pw_{new})$.
Step 2: Replace the original R with this new one, R_{new}.

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, and impersonation attacks. Detail analyses and how the scheme satisfies resisting the above-mentioned attacks are further demonstrated.

Replay attacks

A replay attack is a kind of network attack in which a valid data transmission is maliciously repeated. This kind of attack is generally done by some machinated adversary, who intercepts the data and transmits it repeatedly. In this scheme, the concept of a time stamp is employed to avoid such attacks. When server S or

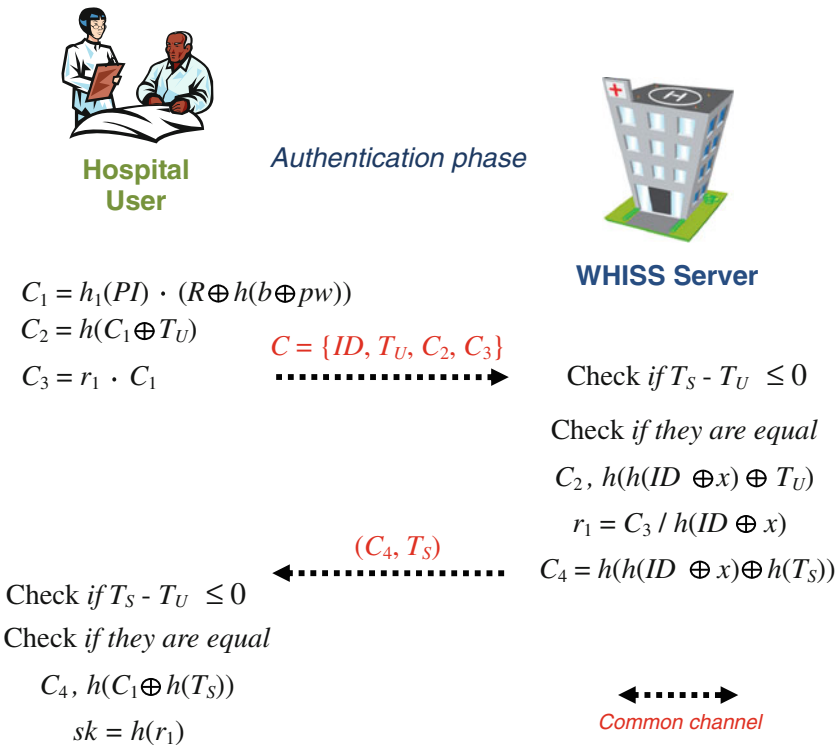

Fig. 3 Authentication phase

user U receives a message, it firstly calculates the difference between the current time T^* and the transmitted time T. And then, it will check whether the difference is smaller than ΔT. If it is, the message is valid; otherwise, the message may be re-sent. Therefore, the replay attack is fruitless.

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 till he is successful. In this scheme, such attack will be recognizable immediately. Suppose an adversary attempts to validate the password of a legal user, a possible password would be guessed to compute some parameters and start to execute the authentication phase. However, the probability of knowing the correct password is only 2^{-k}, where k is the length of the password. According to this scheme, the server can detect abnormality by confirming whether $h(h(ID \oplus x) \oplus T_U)$ is equal to C_2. Generally, when the third guess from the adversary is wrong, it will undoubtedly be kicked out of the system. Therefore, on-line password guessing attacks cannot work here.

Off-line password guessing attacks

For any password-based user authentication scheme, the off-line password guessing attack poses the biggest threat in terms of damage among various types of attacks. An adversary would intercept some transmitted information or generate various self-guessed parameters with brute force to hack the correct password of a specific user. To render this kind of attack being ineffectual, the secret parameters of this scheme such as b and R are protected by the cryptographic hash function and are not revealed to anyone. Assume that an adversary has obtained the following parameters $\{ID, T_U, C_2, C_3\}$ in the login status; however, without b, R, and $h_1(PI)$, the information related to the password will not be obtained by C_2 or C_3. Therefore, off-line password guessing attacks can be withstood.

Stolen-verifier attacks

Stolen-verifier attacks mean that a machinated internal member can steal or modify the passwords or the verification tables of users stored in a server's database. In this scheme, such information is not necessary on the server side as the server can conduct the mutual authentication through its secret number x. Therefore, it is impossible for internal members to steal or modify the passwords. This attack is considered meaningless.

Server spoofing attacks

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

Table 3 Comparison with other related schemes

	Lee-Chiu (2005)	Lu et al. (2008)	Liu et al. (2008)	Xu et al. (2009)	Hsiang-Shih (2009)	Wang et al. (2009)	Our Proposal
Computational operations in registration phase	2H+1E	1H	2H+1E	3H+1E	3H	2H	4H+1M
Computational operations in authentication phase	4H+1E	3H+3EC	3H+3E	4H+3E	6H	8H	10H+2M
Replay attacks	Yes	Yes	Yes	Yes	Yes	Yes	Yes
On-line password guessing attacks	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Off-line password guessing attacks	Yes	No	Yes	Yes	No	No	Yes
Stolen-verifier attacks	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Server spoofing attacks	No	No	Yes	Yes	Yes	Yes	Yes
Impersonation attacks	No	Yes	No	No	No	Yes	Yes
Smart card extraction problems	No	Yes	No	No	No	No	Yes

H one way hash function operations;

M multiplication operations; *E* exponential operations

EC elliptic curve exponential operations

Yes achieve the prevention of the malicious attacks;

No cannot achieve the prevention of the malicious attacks

Commonly, a conspiring attacker has two ways to successfully spoof users, by obtaining the secret of the server and then impersonating as the server to authenticate with users, or by guessing the user's password and directly perform partial phases in the server to communicate with users without the need of secret values. However, these methods are ineffective in this scheme. The secret value x is never transmitted via a common network channel so 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.

Impersonation attacks

Similar to server spoofing attacks, impersonation attacks indicate that someone masquerades as the other legitimate user to log into a server for acquiring services. Obviously, this situation will not appear in this scheme because the password is protected with cryptographic hash function and random values. An adversary, in fact, can hardly generate and interpret authentication messages correctly without knowing a user's password. Consequently, a person who tends to masquerade as the user and acquire services is barred.

Smart card extraction problem

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. This scheme adopts the set of personal information PI to prevent this situation, i.e., the $h_1(PI)$ would be used at each time when the user logins. Therefore, when attackers attempt to masquerade as legal users to login by computing the transmitted value C_2, they will fail without knowing the PI, even when they have got the users' smart cards and extracted the parameters [$h(.)$, $h_1(.)$, b, R]. Thus, the proposed scheme can resist smart card extraction problems.

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 this scheme, each session key is formed by a temporary value $h(r_1)$. 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 the person to encrypt the significant information during the current communication process. Therefore, it is very difficult for anyone to make use of all the known information so as to calculate any possible previous session keys. It can be declared that this scheme achieves perfect forward secrecy.

Comparison

To display how the proposed password-based user authentication scheme is suitably and efficiently implemented, the following is the comparisons of this scheme with other related schemes, as summarized in Table 3. Clearly, Lu et al.'s [27], Lee-Chiu's [13], Xu et al.'s [29], and Liu et al.'s [30] schemes required some exponential operations which required more calculation time and thus resulted into inefficiency. Wang et al.'s [14], Lu et al.'s [27], Lee-Chiu's [13], Hsiang-Shih's [28], Xu et al.'s [29], and Liu et al.'s [30] schemes, all of them could suffer from insecure attacks, such as off-line password guessing attacks, impersonation attacks, server spoofing attacks, and smart card extraction problems, so that they were not practical for implementation. This scheme, on the contrary, presents not only very low computation costs, but requires only few hashing functions and multiplication computations. With the analysis of the nine security concerns mentioned above, the security to apply the mechanism is assured.

Conclusions

In this paper, a password-based user authentication scheme is proposed for the WHISS. The scheme does not need costly, time-consuming exponential computation and thus presents more efficiency. In terms of security, it also shows the security against various attacks, such as replay attacks, on-line and off-line password guessing attacks, stolen-verifier attacks, and impersonation attacks. By achieving perfect forward secrecy which makes it difficult to crack or modify any previous encrypted documents, together with the above mentioned properties, this scheme is worth implementing.

Acknowledgement The authors would like to acknowledge the work of the members of the Center for Infection Control in National Taiwan University Hospital who assisted us to evaluate the WHISS. The research was in part supported by grants DOH 98-DC-1007 from the Center for Disease Control, Department of Health, Taiwan.

References

1. Richards, M. J., Edwards, J. R., Culver, D. H., and Gaynes, R. P., Nosocomial infections in combined medical-surgical intensive care units in the United States. *Infect. Control Hosp. Epidemiol.* 21:510–515, 2000.
2. Tambyah, P. A., Knasinski, V., and Maki, D. G., The direct costs of nosocomial catheter-associated urinary tract infection in the era of managed care. *Infect. Control Hosp. Epidemiol.* 23:27–31, 2002.
3. Rosenthal, V. D., Maki, D. G., Mehta, A., Alvarez-Moreno, C., Leblebicioglu, H., Higuera, F., et al., International nosocomial infection control consortium report, data summary for 2002–2007, issued January 2008. *Am. J. Infect. Control* 36:627–637, 2008.
4. Rosenthal, V. D., Maki, D. G., and Graves, N., The international nosocomial infection control consortium (INICC): goals and objectives, description of surveillance methods, and operational activities. *Am. J. Infect. Control* 36:e1–e12, 2008.
5. Haley, R. W., Quade, D., Freeman, H. E., and Bennett, J. V., Study on the efficacy of nosocomial infection control (Senic Project) - summary of study design. *Am. J. Epidemiol.* 111:472–485, 1980.
6. Gastmeier, P., Geffers, C., Brandt, C., Zuschneid, I., Sohr, D., Schwab, F., et al., Effectiveness of a nationwide nosocomial infection surveillance system for reducing nosocomial infections. *J. Hosp. Infect.* 64:16–22, 2006.
7. Chung, Y. F., Wu, Z. Y., and Chen, T. S., Ring signature scheme for ECC-based anonymous signcryption. *Comput Stand Interfaces* 31(4):669–674, 2009.
8. Ball, E., Chadwick, D. W., and Mundy, D., "Patient privacy in electronic prescription transfer," *Security & Privacy. IEEE* 1:77–80, 2003.
9. Yee, G., Korba, L., and Song, R., "Ensuring privacy for e-health services," in *Availability, Reliability and Security, 2006. ARES 2006. The First International Conference on*, 2006, pp. 8.
10. Yoon, E.-J. and Yoo, K.-Y., "An efficient password authentication schemes without using the server public key for grid computing," In: H. Zhuge and G. Fox, (Eds.), *Grid and Cooperative Computing - GCC 2005*. vol. 3795, ed: Springer Berlin/Heidelberg, 2005, pp. 149–154.
11. Lamport, L., "Password authentication with insecure communication,". *Commun. ACM* 24:770–772, 1981.
12. Wu, Z. Y., Chung, Y. F., Lai, F., and Chen, T. S., "A password-based user authentication scheme for the integrated EPR information system," *Journal of Medical Systems*, doi:10.1007/s10916-010-9527-7, Available online May 27, 2010.
13. Lee, N.-Y., and Chiu, Y.-C., Improved remote authentication scheme with smart card. *Comput Stand Interfaces* 27:177–180, 2005.
14. Wang, Y.-Y., Liu, J.-Y., Xiao, F.-X., and Dan, J., A more efficient and secure dynamic ID-based remote user authentication scheme. *Comput. Commun.* 32:583–585, 2009.
15. M. Meg, "Strategies for the successful implementation of workflow systems within healthcare: a cross case comparison," in *The 36th Annual Hawaii International Conference on System Sciences*, 2003, pp. 166–175.
16. R. Bunge, S. Chung, B. Endicott-Popovsky, and D. McLane, "An operational framework for service oriented architecture network security," presented at the *Proceedings of the Proceedings of the 41st Annual Hawaii International Conference on System Sciences*, 2008.
17. Pierce, M., Fox, G., Youn, C., Mock, S., Mueller, K., and Balsoy, O., *"Interoperable web services for computational portals," presented at the Proceedings of the 2002 ACM/IEEE conference on Supercomputing*. Baltimore, Maryland, 2002.
18. Lewis, G. A., Morris, E., Simanta, S., Wrage, L., "Common Misconceptions about Service-Oriented Architecture," presented at *the Proceedings of the Sixth International IEEE Conference on Commercial-off-the-Shelf (COTS)-Based Software Systems*, 2007, pp. 123–130.
19. Ko, L.-F., Lin, J.-C., Chen, C.-H., Chang, J.-S., Lai, F., Hsu, K.-P., et al., "HL7 middleware framework for healthcare information system," in *e-Health Networking, Applications and Services, 2006. HEALTHCOM 2006. 8th International Conference on*, 2006, pp. 152–156.
20. Hsieh, S. H., Hsieh, S. L., Weng, Y. C., Yang, T. H., Feipei, L., Cheng, P. H., et al., "Middleware based inpatient healthcare information system," presented at *the Bioinformatics and Bioengineering, 2007. BIBE 2007. Proceedings of the 7th IEEE International Conference on*, 2007.
21. Yang, T. H., Cheng, P. H., Yang, C. H., Lai, F., Chen, C. L., Lee, H. H., et al., "A scalable multi-tier architecture for the National Taiwan University Hospital Information System based on HL7 Standard," presented at *the Proceedings of the 19th IEEE Symposium on Computer-Based Medical Systems*, 2006.
22. Health Leval Seven, "HL7 Standard v2.5," ed, 2003, p. 14.
23. Horan, T. C., Andrus, M., and Dudeck, M. A., CDC/NHSN surveillance definition of health care-associated infection and criteria for specific types of infections in the acute care setting. *Am. J. Infect. Control* 36:309–332, 2008.
24. Tseng, Y.-J., Chen, Y.-C., Lin, H.-C., Wu, J.-H., Chen, M.-Y., and Lai, F., "A web-based hospital-acquired infection surveillance information system," in *Information Technology and Applications in Biomedicine (ITAB), 2010 10th IEEE International Conference on*, 2010, pp. 1–4.
25. Kristof, S., Sofie Van, H., Kristof, T., Kristof, L., Filip De, T., Kirsten, C., et al., "Design of software services for computer-based infection control and antibiotic management in the intensive care unit," in *International Conference on eHealth, Telemedicine, and Social Medicine*, 2009, pp. 87–92.
26. Wu, J.-H., Chen, Y.-C., Hsieh, S. h., Lin, H.-C., Chen, Y.-Y., Cheng, P.-H., et al., "Real-time automated MDRO surveillance system.," presented at the *The 2009 International Conference on Bioinformatics & Computational Biology*, Monte Carlo Resort, Las Vegas, Nevada, USA, 2009.
27. Lu, R., Cao, Z., Chai, Z., and Liang, X., A simple user authentication scheme for grid computing. *International Journal of Network Security* 7:202–206, 2008.
28. Hsiang, H.-C., and Shih, W.-K., Weaknesses and improvements of the Yoon–Ryu–Yoo remote user authentication scheme using smart cards Weaknesses and improvements of the Yoon–Ryu–Yoo remote user authentication scheme using smart cards. *Comput. Commun.* 32(4):649–652, 2009.
29. Xu, J., Zhu, W. T., and Feng, D. G., An improved smart card based password authentication scheme with provable security. *Comput Stand Interfaces* 31(4):723–728, 2009.
30. Liu, J. Y., Zhou, A. M., and Gao, M. X., A new mutual authentication scheme based on nonce and smart cards. *Comput. Commun.* 31(10):2205–2209, 2008.
31. Stallings, W., "Cryptography and network security: principal and practices," *5th Edition. Prentice Hall*, 2010.