JOURNAL OF KOREA MULTIMEDIA SOCIETY, VOL. 10, NO. 12, DECEMBER 2007(pp. 1655-1662)

Integration of PKI and Fingerprint for User Authentication

Sam-Bum Shin[†], Chang-Su Kim[††], Yongwha Chung[†††]

ABSTRACT

Although the PKI-based user authentication solution has been widely used, the security of it can be deteriorated by a simple password. This is because a long and random private key may be protected by a short and easy-to-remember password. To handle this problem, many biometric-based user authentication solutions have been proposed. However, protecting biometric data is another research issue because the compromise of the biometric data will be permanent. In this paper, we present an implementation to improve the security of the typical PKI-based authentication by protecting the private key with a fingerprint. Compared to the unilateral authentication provided by the typical biometric-based authentication, the proposed solution can provide the mutual authentication. In addition to the increased security, this solution can alleviate the privacy issue of the fingerprint data by conglomerating the fingerprint data with the private key and storing the conglomerated data in a user-carry device such as a smart card. With a 32-bit ARM7-based smart card and a Pentium 4 PC, the proposed fingerprint-based PKI authentication can be executed within 1.3*second*.

Keywords: User Authentication, PKI, Biometrics, Fingerprint, Smart Card

1. INTRODUCTION

Traditionally, most people set their passwords based on words or numbers that they can easily remember. This makes these passwords easy to crack by guessing or a simple brute force dictionary attack. Although it is possible and even advisable to keep different passwords for different applications, most people use the same password across different applications. If a single password is compromised, it may open many doors. Long and random passwords are more secure but harder to remember, and result in more system help desk calls for forgotten or expired passwords. Cryptographic techniques such as PKI[1] can provide very long passwords that are not required to be remembered but that are in turn protected by simple passwords, thus defeating their purpose.

The increasing demand for more reliable and convenient user authentication solutions generates a renewed interest in human identification based on biometric identifiers such as fingerprints, iris, voice and gait. Since biometrics cannot be lost or forgotten like passwords, biometrics have the potential to offer higher security and more convenience for user authentication[2]. For example, it is significantly more difficult to copy, share, and distribute fingerprints with as much ease as passwords. That is, the main advantage of a fingerprint recognition solution is the convenience it provides the users while maintaining sufficiently high accuracy. However, the fingerprint-based recognition has some disadv]antages as well[2-6]. Although fingerprints are distinctive identifiers, they are not secret. People leave latent fingerprints on everything that they touch. Furthermore, a

※ Corresponding Author : Yongwha Chung, Address : (339-700) Seochang 208, Chochiwon, Chungnam, Department of Computer and Information Science, Korea University, TEL : +82-41-860-1343, FAX : +82-41-864-0014, E-mail : ychungy@korea.ac.kr

Receipt date : Jun. 1, 2007, Approval date : Jul. 19, 2007

[†] Interdisciplinary Program of Information Security, Pukyong National University (E-mail : shinarmy@hanmir.com)

[††] Interdisciplinary Program of Information Security, Pukyong National University

[†††] Department of Computer Information, Korea University (E-mail : ychungy@korea.ac.kr)

compromised password can be canceled and a new password can be issued as often as desired, whereas people have only 10 fingerprints on two hands. If a fingerprint is compromised repeatedly, it cannot be replaced eventually. Finally, in principle, a fingerprint template stolen from one application may be used in another application. These issues are important in pervasive computing where the biometric data must be carefully protected because of privacy concerns[7]. However, only limited research has been carried out in this direction[8-11].

Juels[12] proposed a solution called fuzzy vault, and some implementations results for fingerprint have been reported as a possible solution for cancelable fingerprints. For example, Clancy[13] and Uludag[14] proposed a fuzzy fingerprint vault.

In this paper, we present an implementation for a fingerprint-based PKI user authentication using the idea of fuzzy vault. This solution can improve the security of the existing password-based PKI user authentication by protecting the private key with a fingerprint. This solution also can alleviate the privacy issue of the fingerprint data by storing the fingerprint data not in a database, but in a user-carry device such as a smart card. Furthermore, the fingerprint data stored in the user-carry device is conglomerated with the private key, and the private key is released only with the valid fingerprint. We also evaluate the performance of this solution with a smart card.

The rest of the paper is structured as follows. Section 2 explains previous solutions for user authentication, and Section 3 describes the fingerprint-based PKI solution using the fuzzy vault. The performance comparison is given in Section 4, and conclusions are made in Section 5.

2. BACKGROUND

2.1 Password-based PKI

For the purpose of explanation, we simply assume that both the user(client) and the server know the public keys of the server and the user, respectively. In Fig. 1, for example, user can verify the remote server by checking some information generated by the server's private key with the server's public key, and vice versa[1]. However, most implementations protect the user's or the server's private key with a password, and most people set their passwords based on words or numbers that they can easily remember. Long and random passwords are more secure but harder to remember, which prompts some users to write them down in accessible locations. Note that, for higher security, the enrolled password and the encrypted private key are stored in a user-carry device.

2.2 Fingerprint-based User Authentication

To solve the above problems with a simple password, biometric-based user authentications have been proposed[2]. For example, a typical fingerprint-based authentication solution has two phases: *enrollment* and *verification*. In the off-line enrollment phase, an enrolled fingerprint image is preprocessed, and features, called as *minutiae*, are extracted and stored in a remote server. In the on-line verification phase, the similarity between the enrolled template minutiae and the input minutiae is examined.

Pre-Processing refers to the refinement of the fingerprint image against the image distortion obtained from a fingerprint sensor. *Extraction* refers to the extraction of minutiae in the fingerprint image. After this step, some of the minutiae are

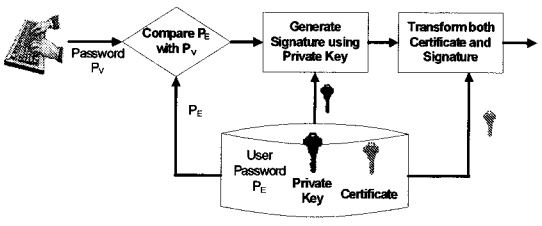


Fig. 1. Illustration of the Typical PKI-based Authentication Solution.

detected and stored into a template file. A minutia can be specified by its coordinates, angle, and its type(ending/ bifurcation).

Based on the minutiae, the input fingerprint is compared with the template fingerprint. Actually, *Match* is composed of alignment stage and matching stage. In order to match two fingerprints captured with unknown direction and position, the differences of direction and position between two fingerprints are detected, and alignment between them needs to be accomplished. Finally, a matching score is computed. Fig. 2 shows the procedures of the typical fingerprint-based user authentication solution.

Although this typical fingerprint-based solution may solve some problems of a simple password, it also has several problems. First, passwords are exactly the same during different authentication attempts, whereas fingerprint images are rarely identical during various acquisitions. This characteristics of fingerprints prevents the same hash from being obtained from different instances of the same finger. Therefore, a fingerprint cannot replace the password shown in Fig. 1. That is, a fingerprint cannot be used with the typical cryptographic hash functions, and a private key cannot be protected by the fingerprint since the fingerprint itself cannot be protected by the hash function. Thus, a user who cannot protect his private key with his fingerprint needs to send his fingerprint directly to the server for client authentication. On the contrary, it is unreasonable for the server to send the fingerprint of the server manager to the user for server authentication. In the sense that a user cannot verify the remote server, this solution can provide the unilateral authentication only. Of course, the server authentication is possible once we assume the private key of the server to be secure, and this will be explained later in Section 3. In principle, however, current fingerprint-based authentication solutions do not match well with PKI because they do not provide reasonable mechanism to protect both the fingerprint and the private key simultaneously.

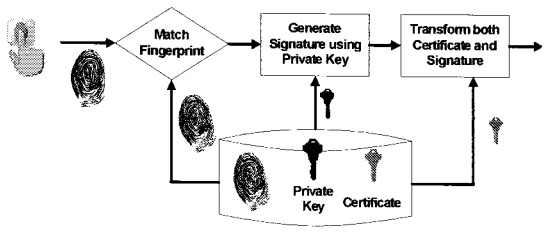


Fig. 2. Illustration of the Typical Fingerprint-based Authentication Solution.

Second problem with the typical fingerprint-based solution is related with the fact that the stored fingerprint template should be protected. Although the server manager tries to protect the fingerprint template, it can be compromised. A compromised password can be canceled and a new password can be issued as often as desired. However, people have only 10 fingerprints at most, and fingerprint cannot be replaced eventually.

Finally, in principle, a fingerprint template stolen from one application may be used in another application. These issues are important in pervasive computing where the biometric data must be carefully protected because of privacy concerns[1-5]. However, only limited researches have been carried out in this direction.

2.3 Protection of Fingerprint Information

Ratha[7] introduced the term "cancelable biometrics" to protect biometric templates. Designing cancelable biometrics had many objectives. First, a cancelable template stored in a database of certain application cannot be used as a template in another application. Second, if a database record(a fingerprint template) is compromised, a new database record can be issued(just like a new password can be issued). Finally, altering a database record(replacing a fingerprint template) is unfeasible because the template can be digitally signed by the issuer, or some privileged information(*e.g.*, an encryption key) can be stored in the template in such

a way that it can be released only through biometric recognition.

Davida[8] suggested on-line biometric authentication which moved biometric data from a central server into a signed form on a portable storage device, such as a smartcard. Their system was essentially a PKI-like environment that did local fingerprint matching. While they address the key management issues, the basic premise is still that of local fingerprint matching.

Tulyakov[9] proposed a method of hashing fingerprint minutiae and performing fingerprint identification using the hashed fingerprint data. Their method is a different approach to implement the cancelable biometrics. However, there are some difficulties in producing good match scores, and setting the match thresholds.

There were other innovative, yet similar methods that did not perform biometric matching. First, Soutar[10] proposed a key binding algorithm in an optical correlation-based fingerprint matching system. This algorithm binds a cryptographic key with the user's fingerprint images at the time of enrollment. The key is then retrieved only upon a successful authentication. However, authors do not explain how much entropy is lost at each stage of their algorithm, and also assume that the input and database template fingerprint images are completely aligned. A second paper[11] has a similar theoretical foundation to this work, but aims toward a completely different application. Here, Monrose attempted to add entropy to users' passwords on a computer system by incorporating data from the way in which they type their passwords. Since the biometric being used here so radically different from fingerprints, their results are not applicable to our work, and Monrose also assumed the aligned biometrics.

Recently, Juels[12] proposed a scheme called *fuzzy vault*. In the fuzzy vault scheme, the secret k is locked by a user's biometric(set A) using a probabilistic LOCK function, resulting in a vault V_A. The corresponding decryption algorithm UNLOCK takes as input a vault V_A, and a decryption biometric(set B) and outputs k if B is close enough to A, or null, otherwise. The authors argued that in a minutiae-based fingerprint matching systems, if a minutiae template is augmented with a larger number of chaff points that constitute random noise, the secrecy of the fingerprint features as well as the secret k is strengthened. Note that the biometric template size increases as a result of introduction of a large number of false features and the accuracy of the fingerprint recognition might be affected.

Based on the fuzzy vault, some implementations results for fingerprint have been reported. For example, Clancy[13] and Uludag[14] proposed a *fuzzy fingerprint vault*. Based on the fuzzy vault, we can improve the security of the existing PKI-based authentication by protecting the private key with a fingerprint.

3. INTEGRATION OF PKI AND FINGERPRINT-BASED USER AUTHENTICATION

To explain the fingerprint-based PKI user authentication solution, we describe the fuzzy vault in more detail. Alice can place a secret value m in a vault and lock it using an unordered locking set $\boldsymbol{L}$. Bob, using an unordered unlocking set $\boldsymbol{U}$, can unlock the vault only if $\boldsymbol{U}$ overlaps with $\boldsymbol{L}$ to a great extent. The procedure for constructing the fuzzy vault is as follows: Secret value m is first encoded as the coefficients of some degree k polynomial in x over a finite field $GF(q)$. This polynomial $f(x)$ is now the secret to protect. The locking set $\boldsymbol{L}$ is a set of t values $l_i \in GF(q)$ making up the fuzzy encryption key, where $t > k$. The locked vault contains all the pairs $(l_i, f(l_i))$ and some large number of chaff points (a_j, β_j), where $f(a_j) \neq \beta_j$. After adding the chaff points, the total number of items in the vault is r.

In order to crack this solution, an attacker must be able to separate the chaff points from the legitimate points in the vault. The difficulty of this operation is a function of the number of chaff points, among other things. A legitimate user should be able to unlock the vault if they can narrow the search space. In general, to successfully interpolate the polynomial, they have an unlocking set $\boldsymbol{U}$ of t elements such that $\boldsymbol{L} \cap \boldsymbol{U}$ contains at least k + 1 elements. The details of the fuzzy vault can be found in [12].

Like the typical PKI-based authentication solution, we can implement the mutual authentication using fuzzy fingerprint vault. To solve the security problems of the typical PKI-based solution, we protect the private key with user fingerprint instead of the user password. Note that, as explained in Section 2.2, the typical fingerprint-based solution supports not the mutual authentication, but the unilateral authentication.

Unlike the typical fingerprint-based solution, the proposed authentication solution matches well with PKI because the typical PKI-based and proposed solutions execute the same tasks after reconstructing the private key as shown in Fig. 3. The private key is combined with the fingerprint template, and only the authorized user can access the private key from the *conglomerate*(*i.e.*, private key/fingerprint template) data by providing the valid fingerprint. For example, the proposed solution extracts an unlocking set from the fingerprint image of the user, and then reconstructs the private key by performing the unlocking operation.

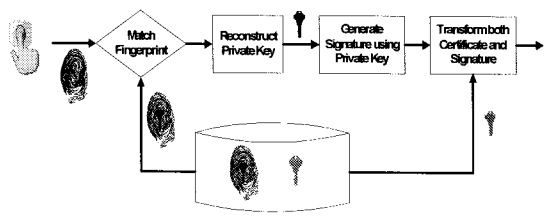


Fig. 3. Illustration of the Fingerprint-based PKI Authentication Solution using Fuzzy Vault.

In the fingerprint matching of the unlocking operation, unlocking set extracted from verifying fingerprint is compared with enrolled vault that conglomerates the private key with the fingerprint template.

To alleviate the privacy issue of the fingerprint data further, we consider storing the conglomerate data not in a centralized database, but in a user-carry device such as a smart card. However, due to the limited computing power of the smart card, the match fingerprint, the reconstruct private key, and the generate signature using private key operations shown in Fig. 3 may not be executed in real-time on the resource-constrained smart card.

Note that, the fuzzy fingerprint vault itself can alleviate the privacy issue of the fingerprint data. That is, the fuzzy fingerprint vault can provide the concept of the cancelable biometrics[10] where the biometric data is not stored in a raw form, but in a non - invertible transformed version of the original biometric data. Thus, even if the storage is compromised, the original biometric data remains safe. Cancelable biometrics also provide a higher level of privacy by allowing many templates for the same biometric data and hence non-linkability of user's data stored in different databases.

As in the scenario of integrating fingerprint with a smart card[15], we consider two scenarios of integrating the fuzzy fingerprint vault with a smart card. The Store-on-Card is used only as a storage device to store the conglomerate data. For example, in a fuzzy fingerprint vault-based Store-on-Card, the conglomerate data stored in the smart card needs to be released into an external card reader or a PC and the three operations are executed outside the smart card. In the Match-on-Card scenario, however, the match operation is performed by the in-card processor, not the external card reader to heighten the security level.

Depending on the integrating scenarios, the security level and the required system resources,

such as the processing power and the memory size, are different. However, there is an open issue of integrating fingerprint verification into the smart card because of its limited resources. In the following, we will compare the execution times of the two integrating scenarios. Note that, because of high cost, we do not consider the System-on-Card scenario where the whole three operations are executed within the smart card and the maximum security can be obtained by removing the risk of any data leaking out.

4. PERFORMANCE COMPARISON

To evaluate the effectiveness of the proposed solution, we compare the typical PKI-based, the typical fingerprint-based, and the proposed authentication in terms of security, privacy, and scalability(see Table 1). The proposed authentication has a good security because it provides the mutual authentication and the private key can be protected by the fuzzy fingerprint vault. Also, the proposed authentication has a good privacy because the fuzzy fingerprint vault is distributed to each user and can be canceled and reissued.

To analyze the feasibilities of the two scenarios(*i.e.*, match-on-card, store-on-card) of embedding the fingerprint-based PKI into the smart card, we compare the execution times of the two scenarios on a 32-bit ARM7-based smart card and a Pentium 4(2GHz) PC. In the performance comparison, 128-bit AES, 1024-bit ECC, and SHA-1[1] are used as our symmetric encryption algorithm, digital signature algorithm, and hash algorithm, respectively.

In the store-on-card scenario, extracting features for unlocking operation(225*ms*), unlocking with the 300 chaff minutiae added(1093*ms*), generating a signature(28.123*ms*) are done by the PC. Thus, the total time for the store-on-card is about 1.3*second*. On the contrary, the execution time for the match-on-card scenario is 57*seconds* on the combination of the smart card and the PC. Thus, with the current smart card technology, the match-on-card scenario cannot be executed in real-time. Note that, we ignore the communication times for the fingerprint sensor, the smart card, and the server in both scenarios.

5. CONCLUSIONS

The use of biometrics in user authentication solutions is very promising. However, without adequate security considerations, the compromise of such biometrics may result in making them being useless for the user forever. In this paper, we presented a solution to provide mutual authentication by using the idea of fuzzy vault. By integrating the fuzzy fingerprint vault with the existing PKI-based authentication solutions, we can improve the security of the existing PKI-based authentication by protecting the private key with a

Table 1. Comparison of the Three Authentication Solutions.

Authentication	Security	Privacy	Scalability
Typical PKI-based	Bad (Private Key protected by Password)	Good (Password managed by Client)	Good (Decentralized)
Typical Fingerprint-based	Bad (Unilateral Authentication, Need Key Distribution)	Bad (Fingerprint Template managed by Server)	Bad (Centralized)
Proposed	Good (Private Key protected by Fuzzy Fingerprint Vault)	Good (Fuzzy Fingerprint Vault managed by Client)	Good (Decentralized)

fingerprint, rather than with a simple password. Also, the solution can alleviate the privacy issue of the fingerprint data by storing the fingerprint data not in a database, but in a user-carry device.

To evaluate the effectiveness of the solution, we compared the typical fingerprint-based, the typical PKI-based, and the proposed authentication in terms of security, privacy, and scalability. Also, we compared the execution times of the match-on-card and the store-on-card scenarios to embed the proposed authentication into a smart card. With the current smart card technology, the store-on-card is the only possible scenario for real-time execution.

REFERENCES

[1] W. Stallings, *Cryptography and Network Security: Principles and Practice*, Prentice Hall, 2003.

[2] D. Maltoni, D. Maio, A. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*, Springer, 2003.

[3] R. Bolle, J. Connell, and N. Ratha, "Biometric Perils and Patches," *Pattern Recognition*, Vol. 35, pp. 2727-2738, 2002.

[4] S. Prabhakar, S. Pankanti, and A. Jain, "Biometric Recognition: Security and Privacy Concerns," *IEEE Security and Privacy*, pp. 33-42, 2003.

[5] U. Uludag, S. Pankanti, S. Prabhakar, and A. Jain, "Biometric Cryptosystems: Issues and Challenges," *Proc. of IEEE*, Vol.92, No.6, pp. 948-960, 2004.

[6] B. Schneier, "The Uses and Abuses of Biometrics," *Communications of the ACM*, Vol.42, No.8, pp. 136, 1999.

[7] N. Ratha, J. Connel, and R. Bolle, "Enhancing Security and Privacy in Biometrics-based Authentication Systems," *IBM Systems Journal*, Vol.40, No.3, pp. 614-634, 2001.

[8] G. Davida, Y. Frankel, and B. Matt, "On Enabling Secure Applications through Off-Line Biometric Identification," *Proc. of Symp. on Privacy and Security*, pp. 148-157, 1998.

[9] S. Tulyakov, F. Farooq and V. Govindaraju, "Symmetric Hash Functions for Fingerprint Minutiae," *LNCS 3687 - Proc. of ICAPR 2005*, pp. 30-38, 2005.

[10] C. Soutar, D. Roberge, S. Stojanov, R. Gilroy, and B. Kumar, "Biometric Encryption Enrollment and Verification Procedures," *Proc. of SPIE*, Vol.3386, pp. 24-35, 1998.

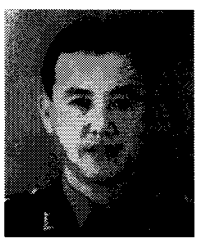
[11] F. Monrose, M. Reiter, and S. Wetzel, "Password Hardening based on Keystroke Dynamics," *Proc. of ACM Conf. on Computer and Comm. Security*, pp. 73-82, 1999.

[12] A. Juels and M. Sudan, "A Fuzzy Vault Scheme," *Proc. of Symp. on Information Theory*, pp. 408, 2002.

[13] T. Clancy, N. Kiyavash, and D. Lin, "Secure Smartcard-based Fingerprint Authentication," *Proc. of ACM SIGMM Multim., Biom. Met. & App.*, pp. 45-52, 2003.

[14] U. Uludag, S. Pankanti, and A. Jain, "Fuzzy Vault for Fingerprints," *LNCS 3546 - Proc. of AVBPA*, pp. 310-319, 2005.

[15] G. Hachez, F. Koeune, and J. Quisquater, "Biometrics, Access Control, Smart Cards: A Not So Simple Combination," *Proc. 4th Working Conf. on Smart Card Research and Advanced Applications*, pp. 273-288, 2000.



Sam-Bum Shin

1992. 2. Korea National Open Univ.(BS)
1996. 8. Dongguk Univ.(MS)
1983~Current Army colonel
Research Interest : Security, USN/RFID

Chang-Su Kim

1984. 2. Ulsan Univ.(BS)
1986. 2. Chungang Univ.(MS)
1991. 8. Chungang Univ.(Ph.D)
1992~Current Pukyong National Univ.
Research Interest : Operating System, USN/RFID, Urban Disaster Prevention System

Yongwha Chung

1984. 2. Hanyang Univ.(BS)
1986. 2. Hanyang Univ.(MS)
1997. 2. Univ. of Southern California(Ph.D)
1986~2003 ETRI
2003. 9.~Current Korea Univ.
Research Interest : Biometrics, Security