# Secure Fingerprint-Based User Authentication for Lotus Notes

Nalini K. Ratha, Jonathan H. Connell and Ruud M. Bolle
IBM Thomas J. Watson Research Center
P. O. Box 704
Yorktown Heights, NY 10598
{ratha, jconnell, bolle}@us.ibm.com

**Abstract**

*Fingerprints have been used to identify people for several decades. With the advent of low cost inkless fingerprint scanners and the ample compute power available in client workstations, biometrics in general, and fingerprints in particular, are being considered for many secure authentication applications. Lotus Notes is a groupware product supporting e-mail, calendar management, workflow, and, perhaps more importantly, (shared) database access and management. Because of this wide spectrum of capabilities that allows true collaborative computing, privacy and security are of primary importance in such groupware applications. Lotus Notes has a strong reputation in this regard. In this paper, we describe the architecture of a system that integrates the intrinsic high security of Notes with the conveniences of fingerprints for client authentication. We discuss several design challenges that had to be addressed to achieve a successful product level system design and development. A demo of the finished system is also available.*

## 1 Introduction

Security is critical in many application, more so in groupware applications that involve work groups and multiple users. There are four primary issues related to security:

- Integrity: The information is accurate, complete and consistent. When the information is transmitted over a network it remains unchanged.
- Privacy: The information is accessible to only those who are authorized. When transmitted information over a network it is accessible to the sender and the receiver only.
- Authenticity: The receiver is assured that the information was truly created by the sender or the specified author. Similarly, the sender is assured that the receiver is genuinely who she or he claims to be.
- Non-repudiation: The transactor cannot deny that the information was created or sent by her or him.

For critical applications such as health care and finances, all four of these security issues are important and need to be carefully addressed. Only authorized users should have access to the applications and related data. For each particular application or database, typically an "access control list" of authorized users with different authorization levels for each user is specified. In more centralized applications, access control is handled by either a traditional userid/password, or other more sophisticated access control mechanisms such as one-time password generators or smart tokens. However, when applications are distributed over several geographical regions and over multiple vendor-based servers interconnected through public networks such as the Internet, the task of securing applications and data becomes extremely complex. Additionally, the maintenance of user IDs

and passwords also becomes in intricate problem, especially in view of the above-mentioned non-repudiation issue.

In the modern networked society, there is an ever-growing need to positively determine or verify the identity of a person. Where authorization is necessary for an action, be it picking up a child from daycare or boarding an aircraft, this authorization is usually vested in a single individual or a class of individuals. There are a number of methods of verifying identity that have been adopted by society or automated systems. These are summarized in Table 1. The existing methods can be grouped into three classes [13]: (i) possessions (what you have); (ii) knowledge (what you know); and, (iii) biometrics (unique personal traits).

Biometrics is the science of identifying or verifying the identity of a person based on physiological or behavioral characteristics. Physiological characteristics include fingerprints and facial appearance. Behavioral characteristics are actions carried out by a person in a unique way. They include signatures and voiceprints, though these are naturally dependent on physical characteristics as well. It is important that the behavioral characteristics must be insensitive to variations due to the state of health, mood of the user, or the passage of time. Similarly, the measured physiological characteristics should remain constant over time.

Often, the three identification methods in Table 1 are used in combination. The possession of a key is a physical conveyor of authorization; a password plus a user ID is a purely knowledge-based method of identification; an ATM card is a possession that requires knowledge (PIN) to carry out a transaction; a passport is a possession that requires biometric verification (passport photo).

Early automated authorization and authentication methods relied only on possessions and knowledge. There are several well-known problems associated with these methods that restrict their use and the extent to which they are trustworthy. The problem is that these methods verify attributes that only indirectly indicate the presence or absence of a given person. Most importantly, the problems are: (i) possessions can be lost, forged or easily duplicated; (ii) knowledge can be forgotten; (iii) both knowledge and possessions can be shared or stolen. Consequently, repudiation is easy. That is, it is easy to deny that a given person carried out an action because only the possession or knowledge is checked and these are loosely coupled to the person's identity. Clearly, this is unacceptable in applications such as high-security physical access control, bank account access, and credit card authentication.

The science of biometrics provides an elegant solution to these problems by positively verifying the identity of the individual. For contemporary applications, biometric authentication is automated to eliminate the need for human verification, and a number of new biometrics have been developed, taking advantage of increasing understanding of the human body and human actions, and advances in sensing techniques [8]. Newer physiological biometric authentication technologies that have been developed include iris patterns, retinal images, and hand geometry; newer behavioral biometrics technologies (although still very much in the research stage) are gait and keystroke patterns. The first step in an automatic biometrics is enrollment of the user (like the registration of a password). After this, the user can be a verified many times.

In this paper, we describe the architecture of a biometrics-based secure authentication implementation for Lotus Notes – a groupware product that supports e-mail, calendar scheduling, and distributed database management. In particular, we address the issues of integrating a fingerprint-based authentication scheme with the existing security infrastructure. Contrary to what may be the common belief, one cannot just install any commercially available fingerprint verification system and reflect the security features in the application. We will discuss the subtle issues involved in integration and demonstrate our solutions to various problems. For instance, one has to be particularly concerned with the security of the fingerprint templates as well as usability issues, such as an auto-detection of the finger on the scanner and multiple finger-based enrollments. Furthermore, a dynamic selection scheme for deciding a threshold for matcher score is incorporated to overcome variability in the quality of fingerprints actually obtained under field conditions. The target application, Notes, also comes with its own specific set of problems. One is that the system allows simultaneous access from different clients as long as

| Method | Examples | Comments |
|---|---|---|
| What you know | User ID, password, PIN | Can be forgotten<br>Easily shared<br>Many passwords are easy to guess |
| What you have | Cards, badges, keys | can be lost or stolen<br>Easily shared<br>Can be duplicated |
| What you know and what you have | ATM + PIN | PIN is a weak link<br>Writing PIN on card<br>Easily shared |
| What you are | Fingerprint, face, ... | Non-repudiable authentication |

Table 1: A categorization of identification technologies.

the used ID file is resident on the connected workstation. This posed a new problem in the overall design and we describe the control flow of the solution we adopted to overcome this difficulty. A successful product-level demo is available based on the described design.

The paper is organized as follows. A generic biometrics-based authentication approach is presented in Section 2. Section 3 describes the baseline fingerprint technology. Section 4 provides a brief introduction to Lotus Notes. In Section 5, the integration issues are enumerated, while specific steps implemented in our design are discussed in Section 6. We give conclusions in Section 7.

## 2 Pattern recognition-based biometrics systems

We can model biometric system as a generic pattern recognition system as shown in Figure 1. The input subsystem consists of a special sensor needed to acquire the biometric signal. Reliable acquisition of the input signal is a challenge for sensor designers, especially in light of varying environmental situations as well as interpersonal and intrapersonal variations. The signal in its raw form contains the required identifying information, but often hidden among much irrelevant information. Invariant features are extracted from the signal for representation purposes during feature extraction (see Figure 1). In the enrollment process, a representation (called template) of the biometrics, based on these features, is stored by the system. The matching subsystem in Figure 1, accepts the query and reference templates and returns the degree of match or mismatch as a score, i.e., a similarity measure. In a final decision step, this score is compared to a decision threshold to declare the comparison a match or mismatch.

The overall performance of such a system depends on the performance of each of the subsystems. In addition, the system designer has to focus on efficient storage and retrieval as well as error-free transmission and possible encryption and decryption of the result and intermediate signals.

To assess the performance of a biometric system, we analyze it in a hypothesis-testing framework. Let $B'$ and $B$ denote biometrics, e.g., two fingers (identities–differenet identities or the same identity). Further, let the stored biometric sample or template be pattern $P' = S(B')$ and the acquired one be pattern $P = S(B)$. Then, in terms of hypothesis testing, we have the null and alternative hypotheses, respectively:

$$H_0 : \quad B = B', \quad \text{the claimed identity is correct} \tag{1}$$

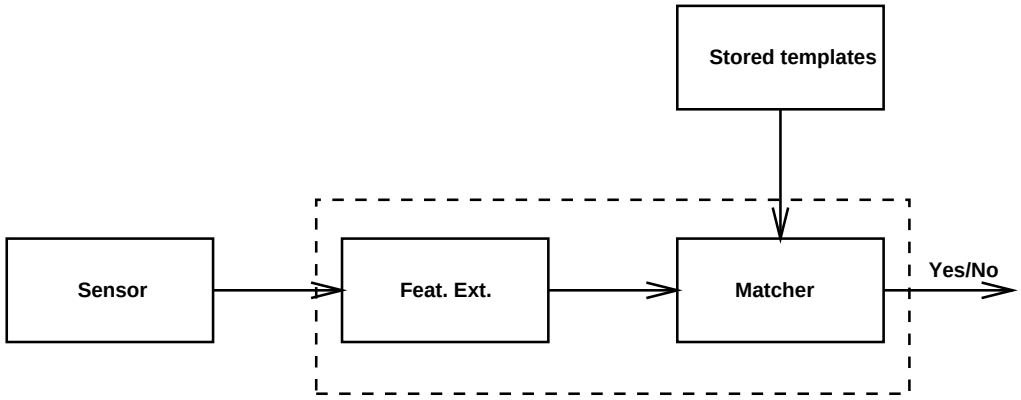


Figure 1: The stages in a generic biometrics system.

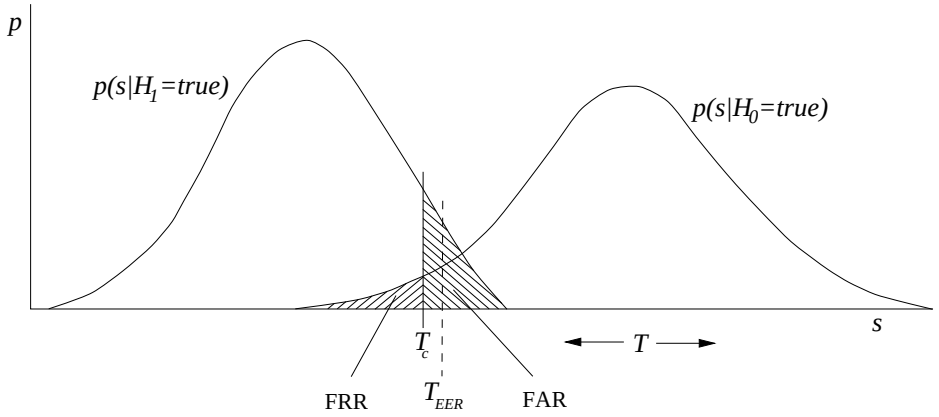


Figure 2: Impostor and genuine distributions with classification error definitions.

$$H_1: \quad B \neq B', \quad \text{the claimed identity is } \textit{not} \text{ correct.}$$

Typically some similarity measure $s = Sim(P, P')$ is defined and $H_0$ is decided true if $s \geq T_d$ while $H_1$ is decided true if $s < T_d$, with $T_d$ a decision threshold. (Some systems use the opposite: a distance or dissimilarity measure. Without loss of generality we assume a similarity measure throughout.) The measure $s$ is also referred to as a *score*. When $B = B'$, $s$ is referred to as a *match score* and $B$ and $B'$ are called a *mated pair* or *matched pair*. When $P \neq P'$, $s$ is referred to as a *non-match score* and $B$ and $B'$ are called a *non-mated pair*.

A measure of "goodness" $d'$ (d-prime) of a matcher can be defined in terms of parameters of the PDFs as [4]:

$$d' = \frac{\mu_1 - \mu_2}{\sqrt{(\sigma_1^2 + \sigma_2^2)}}. \tag{2}$$

This measure was originally developed to measure the separability of two normal (or at least symmetric) distributions. This measure is fine if one insists on a *single* number that expresses the quality of a matcher, we prefer the use of ROCs as described below.

For expression 1, deciding $H_0$ when $H_1$ is true gives a "false acceptance." Similarly, deciding $H_1$ when $H_0$ is true results in a "false rejection." The False Accept Rate (FAR) (proportion of non-mated pairs resulting in false acceptance) and False Reject Rate (FRR) (proportion of mated pairs resulting in false rejection) together characterize the accuracy of an authentication system for a given decision threshold. The Equal Error Rate

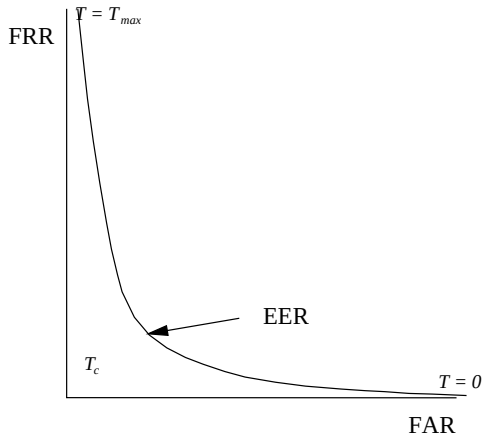


Figure 3: Receiver Operating Curve (ROC).

(EER) is the point at which threshold ($T_{EER}$), where FRR = FAR, *i.e.*, where the areas marked under the two curves in Figure 2 are equal.

Rather than showing the error rates in terms of probability densities as in Figure 2, it is more desirable to report system accuracy in terms of a Receiver Operating Curve (ROC) [5, 14]. A ROC is a mapping $T_d \rightarrow (\mathrm{FAR}, \mathrm{FRR})$,

$$\mathrm{ROC}(T_d) = (\mathrm{FAR}(T_d), \mathrm{FRR}(T_d)),$$

as shown in Figure 3. Note that in a typical recognition system, all the information contained in the PDFs is also contained in the ROC. The ROC just more explicitly shows the tradeoff between FAR and FRR as the decision threshold is varied (the system can operate at any point along the curve).

# 3 Automated Fingerprint Identification Systems

Fingerprints could be called the "mother of all biometrics" and, certainly are the most widely used biometric. The advance in inkless fingerprint-scanning technology, coupled with the exponential increase in processor performance, has taken fingerprint recognition beyond criminal identification applications. Consequently, civilian applications such as access control, time and attendance tracking, and computer user login are quickly emerging.

Over the last decade, many novel techniques have been developed to acquire fingerprints without the use of ink. These scanners are known as "livescan" fingerprint scanners. The basic principle of these inkless methods is to sense the ridges on a finger, which are in contact with the surface of the scanner. Currently livescan acquisition systems are based on four technology types:

- **Frustrated total internal reflection (FTIR) and other optical methods** (e.g., [6]): This technology is by far the oldest livescan method. A camera acquires the reflected signal from the inner surface of a prism as the subject touches the outer surface. Where there is no contact, all the light is reflected off the surface. Where contact occurs some light leaks out due to a similar index of refraction. A typical area of 1" $\times$ 1" is converted to a 500 dpi image using a CCD or CMOS camera. Many variations of this principle, such as use of tactile sensors instead of a prism and the use of a holographic element [12], are available. The main usability issue with these scanners is that the reflected light is a function of skin characteristics. If the skin is either too wet or dry, the fingerprint impression can appear "saturated" or fragmentary, respectively, and thus be hard to process.

(a) (b)

Figure 4: A fingerprint image (a); thinned ridges with minutia features marked (b).

- **CMOS capacitance** [9]: The ridges and valleys of a finger create different charge accumulations when the finger touches a CMOS chip grid. With suitable electronics, the charge is converted to numerical pixel values. Normally at 500 dpi, these scanners provide about $0.5" \times 0.5"$ of scan area. This can be a potential problem because fingerprint impressions acquired at different times may have little overlap due to positioning or rotation. These sensors also are affected by the skin dryness and wetness, and have an additional vulnerability due to electrostatic discharge.

- **Thermal** [10]: A pyro-electric material in the sensor measures temperature changes as the finger is swiped over the scanner and produces an image. This technology claims to overcome the dry and wet skin issues in the optical scanners and can sustain higher static discharge. However, the images are somewhat lacking in contrast and subject to mosaicking errors.

- **Ultrasound** [2]: An ultrasonic beam is scanned across the fingerprint surface to measure the ridge depth from the reflected signal. Skin conditions such as dryness, wetness, and oil on the skin have very little effect and the images reflect the actual ridge topography better. However, these units tend to be very bulky and require a longer scanning time than optical or CMOS scanners.

Recently, completely non-contact [1] fingerprint scanners have been announced. These are not only less intrusive but also avoid the many problems related to touch-based sensing methods, such as elastic distortion of the skin pattern, and wetness and dryness.

Once a fingerprint image is acquired by some means, it must be analyzed and invariant properties need to be identified. The most commonly used fingerprint features are ridge bifurcations and ridge endings, collectively known as *minutiae*, which are extracted from the acquired image. Figure 4 shows a fingerprint image and a thinned version of the ridges. Overlaid on these are the minutiae point features.

The feature extraction process starts by examining the quality of the input fingerprint impression. Then, virtually every published method of feature extraction (e.g., [11, 15]) computes the orientation field of the fingerprint image that reflects the local ridge direction at every pixel. This local ridge orientation is also commonly used to tune filter parameters for enhancement and ridge segmentation. From the segmented ridges, a thinned image is computed to locate the minutiae features. Usually, a minutiae post-processing stage cleans up spurious minutiae resulting from either enhancement, ridge segmentation, or thinning artifacts.

But even after the features have been extracted, the authentication function (matcher) still has to compensate for: (i) translation, (ii) rotation, (iii) missing features, (iv) additional features, (v) spurious features, and, more importantly, (vi) elastic distortion between a pair of feature sets. Variation can also be due to unexpected sources. For instance, storage and transmission of fingerprint images often involves compression and decompression of the image. Standard compression techniques often remove the important high frequency information around the minutiae features and hence can impair recognition. To overcome this, a fingerprint compression scheme called as Wavelet Scalar Quantization (WSQ) has been endorsed by the FBI.

# 4 Lotus Notes Security

Notes is a groupware product from Lotus designed to handle messaging, calendars, and collaborative activities within workgroups based on client-server architecture. Notes integrates essential technologies, among which: (i) e-mail; (ii) shared databases; and, (iii) workflow. E-mail provides communication capabilities. Shared databases provide collaboration possibilities, and workflow supports coordination. These powerful building blocks give Notes useful functions such as shared document management, integrated development environments, support for mobile users, local database replication for off-line access to documents from the network, and synchronization of local copies with the server version. More details of Notes functionality can be obtained from [7]. For all these functions, the security of access is extremely important.

Notes security depends on a certificate that represents the trust between the user and the server. The certificate is stored in a Notes ID file. Notes incorporates a number of cryptographic techniques to establish the trust between the client and the server. When a user is enrolled on a Notes server system, a Notes ID file is created and provided to the user. To use the system, the user needs to make this ID file available on the client workstation. The system is intentionally designed so that it is possible for several distinct clients to use a copy of the same ID file at the same time. The ID file itself contains the server name, connection details, and the public key of the server for authentication between the client and server. Additionally, it contains the user name and password, as well as the relevant private cryptographic keys and their expiration dates. The password is used to first unlock the ID file, and this information, stored in the ID file is used to authenticate both the user and server. In principle, multiple copies of the Notes ID file could have different passwords if the system administrator allows it.

Several anti-spoofing techniques are employed by Notes while reading the password in the dialog box. The first feature being a random number of ”X”s appears for each letter as it is typed by the user. Secondly, a series of hieroglyphic symbols appear on the left side of the box and change dynamically. If the user does not see these figures, they can suspect it to be fake system trying to read the user’s password. Finally, an increasing delay every time a wrong password is typed is used to frustrate any impostors trying to quickly guess many variations of a password. Notes administrative policy also supports saving multiple passwords in an ID file and requiring that $k$ out of $m$ of these be provided for authentication.

The actual authentication task between the Notes client and the Notes server is performed using a series of cryptographic protocols where both the client and server validate each other through challenge and response. The protocol used for authentication is similar to X.509 protocol. The public keys are exchanged through the Notes ID file and random challenges are encrypted using these public keys. A party is validated if it can decrypt the received challenge message and send the correct response. This part of the secure access procedure would remain the same even if the password were replaced with other methods of authentication.

Notice that the issue of non-repudiation is not addressed in the Notes authentication, though a sophisticated protocol is used for establishing trust between client and the server. Our approach is to provide a reliable, accurate positive authentication by replacing the password with a biometric, in particular, with a fingerprint.

# 5 Integration issues

Several issues need to be addressed while integrating an emerging technology like biometrics into a more established technology like groupware. The main issues involved in the integration of a fingerprint-based authentication method are:

- Biometrics independence: The design should be flexible enough to handle other biometrics beyond the one chosen in a particular implementation. In addition, multi-biometrics, i.e., the use of more than one biometric, should be included in the design.
- Sensor independence: In the design, no assumptions should be made about particular sensor characteristics. For enterprise-wide deployments, single-sensor manufacturer dependence is probably are not desirable.
- Development issue: Commercial, off-the-self products are often designed as stand-alone systems, not including a development toolkit. This makes them difficult to integrate into an existing application like Notes.
- Auto-detect and snap: During the input process, the sensor should require very little intervention from the user. The process itself should decide when there is a meaningful and reliable signal to be analyzed and snap the image at that point, or for a limited period from that point, and match only that particular signal.
- Quality: The system should attempt to ensure, or at least measure, the quality of the input automatically. Users typically have no feel for the factors that contribute to good or poor signals, and cannot make suitable adjustments on their own.
- Enrollment issues: Any authentication system requires two steps: one-time enrollment and multiple authentication. Enrollment is the single most important step that can affect the performance of the system. Often, this has to be done in an unsupervised manner in enterprise-wide application like Notes. How many samples of the signal should be acquired and how should they be combined, is an issue here?
- Threshold selection: As observed earlier, the system performance can change significantly depending on the operating point chosen on the ROC. Letting the user decide the threshold or the operating point is not a good idea since they do not understand the full ramifications. At the same time, once cannot have a human expert decide this for every individual enrollee. Employing an automatic threshold-selection process may be desirable.

Having stated these design issues, we describe the solutions employed in our system. There exists a Notes API for C++ and we have employed that along with the IBM fingerprint authentication library API to replace the password-based authentication. We use the industry standard MS Crypto API (MSCAPI) to add extra security where needed.

## 5.1 Scanner interface

As noted earlier, there are many different types of fingerprint scanners. As the fingerprint features used for authentication are coordinates of minutiae, cross-sensor matching deserves special attention. The main issues that affect the design are (i) imaging resolution; and, (ii) pixel-aspect ratio. To handle these variations we create a configuration file that lists these parameters so that the matcher can translate features from each scanner into a canonical spatial reference frame.

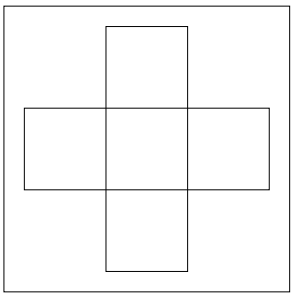

Figure 5: Region used for auto snap.

Score=0.91 Score = 0.44 Score = 0.16

Figure 6: Fingerprint image quality assessment. (a) good; (b) medium; (c) poor.

### 5.2 Auto-snap

A desirable feature during enrollment and verification is to automatically detect the fingerprint and snap the best frame for further processing. We have designed a simple and fast method for detecting a fingerprint based on the following observation. The central part of the acquired image frame has a different gray-scale mean and variance when a finger is present compared to when a finger is not present. Typically without a fingerprint, the average pixel intensity is close to 255 and variance is close to zero. When a fingerprint is present, the mean pixel value is less than 255 (ideally 128 as the ridge and valley pixels are equally distributed in the band shown in the Figure 5) and the variance is close to $128 \times 128$. Moreover, once a finger is present in a particular frame, successive frames for some amount of time should still have the finger present.

Three consecutive images are used to decide if there is a finger present in a frame. Note that on some optical scanners, one can easily create a non-fingerprint image that the system can accept. Once a set of candidate frames has been selected based on their closeness to the ideal characteristics, the quality of the each individual fingerprint image is assessed as described in the next section. The advantage of the auto-snap feature is that it does not require the user to click a mouse button or special key to indicate the presence of a fingerprint. The scanner itself becomes a “button” of sorts.

### 5.3 Image quality assessment

The quality of the acquired image can have a significant impact on the performance of the system. Particularly during enrollment, care should be taken to acquire the best image possible. We use a quality index described in [3] to determine the quality of the fingerprint present. Intuitively, the ideal fingerprint is characterized by

smoothly flowing ridges and valleys with good contrast. The quality index reflects these characteristics and explicitly detects image features indicating that the finger is too wet or dry, or is moving (smudged). A set of good, medium and poor images is shown in Figure 6 along with the image quality scores. To compute the overall quality index, the image is divided up into smaller blocks and for each block an atomic quality is computed by determining consistency in ridge flow directions.

### 5.4 Enrollment issues

To make the system more usable, we acquire templates of two fingers during the enrollment process (preferably from two different hands) with two impressions of each finger. The same acquisition process is adopted for all the four finger impressions using the quality index as described above. The overall process of replacing the passwords is simple. The "change password" Notes API call allows for replacing passwords by other authentication methods. It is assumed that the user gets an initial text password and will then be required to change it at the first login to the Notes system. At that point, the fingerprint-based authentication will replace the password-based method. During enrollment, the user alternates between the primary finger and alternate finger to provide as much variation as possible between the two samples. The enrollment process also checks to ensure that the primary finger and alternate finger are truly different by matching them against each other.

An interesting case occurs when the user has both fingerprint and password access to the system. This might occur, for instance, if his home machine did not have a fingerprint reader.

### 5.5 Threshold selection

The authentication subsystem uses a threshold on the degree of match to decide when there is a match. As stated earlier, the threshold plays a crucial role in determining the two types of system errors. We have developed a novel scheme to dynamically determine the threshold for a person. This adjustment is required because each pair of matching fingerprints does not produce similar matching scores. The matching score largely depends on the image quality, reliability of the features extracted, and the distortion of the features due to skin elasticity. Our threshold is decided as per the following equation based on the pairs of images acquired during enrollment. We have two thresholds, $T_1$ and $T_2$.

$$T_i = \begin{cases} 25 & : \quad matchscore(s_1, s_2) < 35; \\ 0.7 * matchscore(s_1, s_2) & : \quad otherwise. \end{cases}$$

where $s_1$ and $s_2$ are the two fingerprint samples or templates.

### 5.6 Encryption

The use of standard encryption enhances the overall security of the system. The templates extracted during enrollment have to be stored in a secure location as they are keys to the secure authentication. The four templates and associated match thresholds are encrypting using a standard key-based encryption method. The encrypted templates are stored in the template database portion of the user's Notes ID file. The decryption keys are known only to application which decodes the templates as required during authentication.

## 6 Conclusions

Existing methods of automatic authentication involving knowledge or possessions have a number of limitations, particularly that they can be transferred from one person to another. Automated biometrics can address this

problem while also overcoming other problems such as loss and forgery. Automated biometrics, by measuring hard-to-forge characteristics inherent to a person provides a reliable, non-repudiable guarantee of dentity. Recent innovations in hardware and algorithms have meant that the field of biometrics has expanded tremendously, and many applications, not just in security, are being implemented with the use of biometric technology.

Here we have detailed one such commercial application. We have outlined the design issues while integrating a biometrics with another application. In particular, scanner inter-operability, autodet ect and auto snap, image quality assessment, enrollment issues and threshold selection are some of the design decisions we had to make. In addition, we have combined existing encryption methods to provide a secure user-authentication for Lotus Notes. We have also discussed how automated biometric systems in general can be modeled as pattern recognition systems, particularly for evaluating their security performance.

# References


[1] *Non-contact fingerprint scanner: http://www.ddsi-cpc.com/pages/products/cscan300.html*.

[2] W. Bicz, Z. Gurnienny, and M. Pluta. Ultrsound sensor for fingerprints recognition. In *Proc. of SPIE, Vol. 2634, Optoelectronic and electronic sensors*, pages 104–111, June 1995.

[3] R. M. Bolle, S. Pankanti, and Y. Yao. *System and method for determining the quality of fingerprint images*. US Patent number, US5963656, 1999.

[4] J. G. Daugman and G. O. Williams. A proposed standard for biometric decidability. In *CardTechSecureTech*, pages 223–234, Atlanta, GA, 1996.

[5] B. G. et al. Issues in large scale automatic biometric identification. In *IEEE Workshop on Automatic Identification Advanced Technologies*, pages 43–46, Stony Brook, NY, Nov. 1996.

[6] D. T. Follette, E. B. Hultmark, and J. G. Jordan. *Direct optical input system for fingerprint verification*. IBM Technical Disclosure Bulletin: 04-74p3572, April 1974.

[7] I. ITSO. *Lotus Notes and Domino R5.0 security infrastructure revealed*. IBM Corporation, 1999.

[8] A. Jain, R. Bolle, and S. Pankanti, editors. *Biometrics–Personal Identification in Networked Society*. Kluwer Academic Publishers, Boston, 1999.

[9] S. Jung, R. Thewes, T. Scheiter, K. F. Gooser, and W. Weber. A low-power and high-performance cmos fingerprint sensing and encoding architecture. *IEEE Journal of Solid-state Cicuits*, 34(7):978–984, July 1999.

[10] J.-F. Mainguet, M. Pegulu, and J. B. Harris. Fingerchip$^{TM}$: thermal imaging and finger sweeping in a silicon fingerprint sensor. In *Proc. of AutoID 99*, pages 91–94, October 99.

[11] D. Maio and D. Maltoni. Direct gray-scale minutiae detection in fingerprints. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(1):27–40, January 1997.

[12] M. H. Metz, Z. A. Coleman, N. J. Phillips, and C. Flatow. Holographic optical element for compact fingerprint imaging system. In *Proc. of SPIE, Vol. 2659, Optical security and counterfeit deterrance techniques*, pages 141–151, 1996.

[13] B. Miller. Vital signs of identity. *IEEE Spectrum*, 31(2):22–30, February 1994.

[14] W. Peterson, T. Birdsall, and W. Fox. The theory of sigmal detectability. *Transactions of the IRE*, PGIT-4:171–212, April 1954.

[15] N. K. Ratha, S. Chen, and A. K. Jain. Adaptive flow orientation based texture extraction in finger print images. *Pattern Recognition*, 28(11):1657–1672, November 1995.