

# A biometrics-based secure authentication system

Nalini K. Ratha Jonathan H. Connell and Ruud M. Bolle  
IBM Thomas J. Watson Research Center  
30 Saw Mill River Road  
Hawthorne, NY 10532  
{ratha, jhc, bolle}@watson.ibm.com

## Abstract

*Over the last couple of years, there has been a significant surge in the use of biometrics for user authentication applications. Biometrics-based authentication offers several useful advantages over knowledge and possession-based methods such as password/PIN-based systems. When employed in security-critical applications, and more so in unattended remote applications, the biometrics-based authentication systems should be designed to resist different sources of security attacks on the system. In this paper, we present the inherent strengths of a fingerprint-based authentication scheme and also describe the security holes in such systems. A new solution is presented to alleviate one of the weak links in the system.*

## 1 Introduction

Many applications in everyday life require user authentication. The prevailing techniques of user authentication involving passwords and user ids, or identification cards with PINS suffer from several limitations. The main problems with such systems is that the authentication subsystem can be fooled very easily and also there is no way to link the user to the usage of the system. For example, the user id and password can be shared with a colleague. Thus the security of the system is compromised severely. There are many applications where such security lapses cannot be tolerated. It is more difficult to share a biometric of a person with another.

But when biometrics is employed in security critical applications, the hackers will find the weak points in the system and attack the systems at those points. Unlike password systems,

which are prone to password dictionary attacks, the biometrics systems require much more effort. In supervised use of biometrics as an authentication tool, this may not be a concern. But in remote unattended application such as web-based e-commerce applications, hackers will have enough time to make several attempts before giving up and still be unnoticed. Standard crypto techniques will be useful in many ways to prevent a breach of security. But several new type of attacks are possible in the biometrics domain.

In this paper, our goal is to make a study of weak points in a biometrics-based authentication system. Though our analysis is very general, we focus on fingerprints in Section 2. We analyze the power of a minutia-based fingerprint system in terms of probability of a brute force attack being successful. In Section 3, we describe a pattern recognition model of a generic biometrics system to identify the possible attack points. In Section 4, a challenge/response-based method is proposed to address one form of attack.

## 2 Brute force attack

In this section we show the relationship between the number of brute force attack attempts as a function of number of minutiae that are expected to match in the matcher subsystem. Generating all possible images to guess the matching fingerprint image has a much larger search space.

For the purpose of analyzing the “dumb” minutia brute force dictionary attack, we assume the following.

- The system uses a minutia-based method and the number of paired minutiae reflect the degree of match

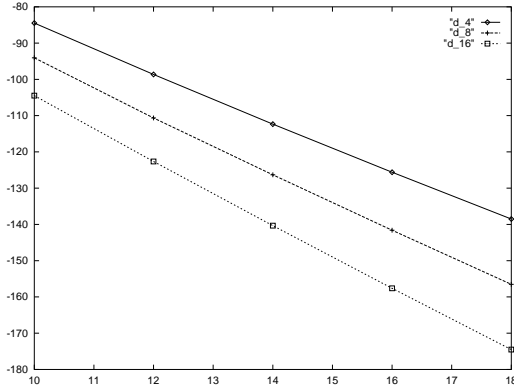


Figure 1: Probability of a successful brute force attack.

- Image size (S) =  $300 \times 300$
- A ridge plus valley spread (T) = 15 pixels
- Total number of possible minutia sites ( $K=(S/T)^2$ ) =  $20 \times 20 = 400$ .
- Number of orientations allowed for the ridge angle at the minutia point (d) = 4, 8, 16
- Minimum number of common minutiae in a query template ( $N_q$ ) = 10, 12, 14, 16, 18

Possible ways to place  $N_q$  minutiae in a possible

$$K \text{ locations} = \binom{K}{N_q};$$

Possible ways to assign directions to each minutia =  $d^{N_q}$ ;

Hence, total possible minutia combinations =

$$\binom{K}{N_q} \times (d^{N_q}); \quad (1)$$

Plugging the values into equation (1), for  $d=4$ ,  $N_q=10$ , the probability of randomly guessing a correct feature set =  $2^{-84.5}$ . The  $\log_2$  of probability of randomly guessing a correct feature set through brute force attack for different values of  $d$  and  $N_q$  is plotted in Figure 1. This is the equivalent number of bits in a fingerprint when considered as a password. This should convince the readers that a brute force attack in the form of a random image or a random template to impersonate the system will require a lot more effort.

Note that the matcher has been assumed to tolerate shifts in minutia points at most by a ridge

and valley pixel width. We did not constrain the angles at every minutia. A more powerful model is under development. If only those minutia patterns are generated that represent “true” fingers by modeling priors and dependence between minutiae, this number could be lower than shown here. The probability of break-ins when good quality fingers are enrolled is of course much smaller than that for poor quality fingerprint images and may be near this theoretical upper bound. The forgoing analysis assumed each fingerprint had exactly  $N_q$  minutiae, only  $N_q$  minutiae were generated and that all of these had to match.

A realistic number is much lower because one can generate more than  $N_q$  minutiae say  $N_{total}$  and some  $N_q$  of them must match some  $N_q$  of the reference fingerprint. This leads to a factor of about  $\left(\frac{N_{total}}{N_q}\right)^2$  or a loss of nearly 64 bits in strength for  $N_q=10$  with  $N_{total} = 50$ .

### 3 Possible attack points

A generic biometric system can be cast in the framework of a pattern recognition system. The stages of such a generic system are shown in Figure 2. There are in total eight possible sources of attack on such systems as described below. Schneier describes many other types of abuses of biometrics in [3].

1. Fake biometric at the sensor: In this mode of attack, a possible reproduction of the biometric being used will be presented to the system. Examples include a fake finger, a copy of a signature, a face mask.
2. Resubmission of old digitally stored biometrics signal: In this mode of attack, an old recorded signal is replayed into the system bypassing the sensor. Examples include presentation of an old copy of fingerprint image or recorded audio signal of a speaker.
3. Override feature extract: The feature extractor could be attacked with a Trojan horse to change it to produce feature sets of choice.
4. Tampering with the feature representation: After the features have been extracted from the input signal, in this mode, they are replaced with a synthesized feature set

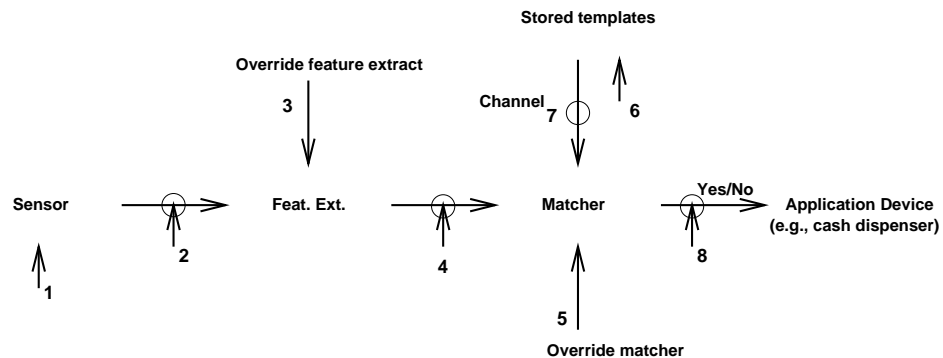


Figure 2: Possible attack points in a generic biometrics-based system.

of choice assuming the representation is known. Often the two stages of feature extraction and matcher are inseparable and this mode of attack is extremely difficult. However, if minutiae are transmitted to a remote matcher (say over the Internet) than this threat is very real. One could snoop on the TCP/IP stack inside the computer and alter certain packets.

5. **Override matcher:** The matcher is attacked to produce the desired result.
6. **Tampering with stored templates:** The database of enrolled templates is available locally or remotely. This database can also be distributed over several servers. The stored template attacker tries to modify one or more templates in the database which could result in at least denial of service for the corrupted template.
7. **Channel attack between stored templates and the matcher:** The templates from the stored database are sent to the matcher through a channel which could be attacked to change the contents of the templates before they reach the matcher.
8. **Decision override:** If the final result can be overridden with the choice of result from the hacker, the final outcome is very dangerous. Even if the actual pattern recognition system had an excellent performance characteristics, it has been rendered useless by a simple exercise of overriding the result.

There exists several techniques to thwart attacks at various points. For instance, sensing finger conductivity or pulse can stop simple attacks at point 1. Encrypted communication channels [2] can eliminate at least remote attacks at point 4. However, even if the hacker cannot get inside the feature extract machine, the system is still vulnerable. The simplest way to stop attacks at points 5, 6 and 7 is to have the matcher and database reside in a secure location. Of course even this cannot prevent attacks in which there is collusion. Cryptography again can help at point 8.

## 4 Challenge/response method

We propose a new method that can handle the attacks of type 2 on the input signal. The motivation of our approach is based on a challenge/response systems. Conventional challenge/response systems are based on challenges to the user. Our approach is based on challenges to the sensor. The sensor is assumed to have enough intelligence to respond to the challenges. Standard cryptographic techniques though mathematically strong are computationally very intensive and would require maintaining a secret key base for a large number of sensors. Moreover, the encryption techniques can not check for liveness of a signal. An old stored image can be given to the encryptor. Similarly a digital signature of a signal does not check for its liveness. We exploit the availability of a large number of image pixels and simple image related challenges that can be posed to the sensor. We assure liveness by challenging the sensor with a pseudo-random challenge. The

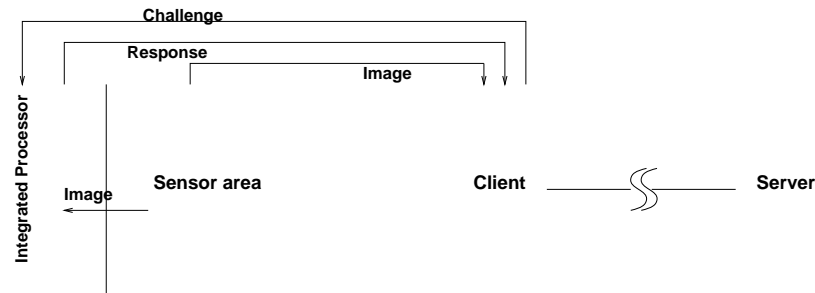


Figure 3: Signal authentication based on challenge/response.

sensor being intelligent responds to this challenge as per the scheme shown in Figure 3.

Our proposed solution works as follows. At the user terminal or system, the transaction gets initiated. The transaction server generates a pseudo-random challenge for the transaction and the sensor. Note that we assume that the transaction server is assumed to be secure. The user system passes the challenge to the intelligent sensor. The sensor acquires a signal at this point of time and computes the response to the challenge. For instance, the integrated processor might be able to compute either of two selectable functions, “x1+” and “x10+”. Bank A might use function “x1+” in all its units, while Bank B might use “x10+” in all of its. Or, alternatively, for even numbered transactions function “x10+” might be used, and for odd numbered transaction “x1+” is used. Hence, the challenge augments the challenge through one or more functions.

The important point is that the response depends on the challenge and the image itself. A typical challenge might be “3, 10, 50”. This would be augmented by function “x1+” by appending all the pixels values of the image (in scan order) to the end of the challenge string. The integrated processor then selects the 3rd, 10th and 50th pixel value from this sequence to generate an output response such as “133, 92, 176”. Other examples of responder functions includes computing a checksum of a segment of the signal, a set of pseudo-random samples, a block of contiguous samples starting at a specified location and with a given size, a hash of signal values, and a specified known function of selected samples of the signal. A combination of these functions can be used to

achieve arbitrarily complex responder functions. The signal as well as the response is transmitted to the server where the response can be verified.

By integrating the responder onto the same chip as the sensor it is just about impossible to inject a fake image (point 2 attack). When the compute power is significant, we can carry out many novel solutions. For example, data hiding in a compressed domain can virtually enhance the performance of the solution significantly. Many silicon fingerprint scanners [1] will be able to exploit the proposed method as they can integrate a processor without much effort.

## 5 Conclusions

As biometrics-based authentication becomes an integral part of overall security, biometrics systems have to be designed to be more robust to attacks from hackers to prevent breakins. We highlighted the five weak points in a generic biometrics systems model. A challenge/response method to authenticate signal from an intelligent sensor has been proposed to alleviate some of the security threats.

## References

- [1] T. Rowley, “Silicon Fingerprint Readers: A solid state approach to biometrics”, Proc. of the CardTech/SecureTech, Orlando, Florida, May 97, Vol. 1, pp. 152–159.
- [2] B. Schneir, “Security pitfalls in cryptography”, Proc. of CardTech/SecureTech, Washington D.C., April 98, Vol. 1, pp. 621–626.
- [3] B. Schneier, “The uses and abuses of biometrics”. Communications of the ACM, August 1999, Vol. 42, No. 8, pp. 136.