BASE: a Proposed Secure Biometric Authentication System

Colby G. Crossingham and Sebastian H. von Solms

Academy For Information Technology
University of Johannesburg
Johannesburg, South Africa

Abstract. The use of biometrics as a secure remote authentication mechanism is hindered by a series of issues. In the case of fingerprints, an attacker can make physical copies of fingerprints by 'lifting' latent fingerprints off a non-porous surface. These lifted fingerprints can be used to illegitimately gain access to an authentication system. Password authentication systems only accept passwords that match 100%, whereas biometric authentication systems match submitted tokens provided they fall within a specified threshold. Without making use of a sufficient encryption scheme, illegitimately replaying a biometric token that has been tweaked slightly could still cause the authentication system to accept the submitted biometric token as a fresh biometric. BASE (Biometric Authentication System Entity), the proposed system, provides a solution to these challenges. As an alternative to current remote authentication systems, the BASE system utilizes a communication protocol that does not make use of encryption, passwords or timestamps, whilst still maintaining the security of the remote authentication process.

1 Introduction

Passwords have long been used as the mainstream mechanism for authentication. Even though there is a widespread popularity with using passwords, it is not difficult to recognize how vulnerable they are. In many cases passwords are forgotten, guessed, written down and even cracked using various brute-force attack techniques [1-3]. The need for more secure authentication mechanisms should be an important concern.In this paper, we are presenting the case of a large automotive- extended enterprise that decided to take action and takes care of its dealers network that are external actors interacting directly with the customers of the EE. We first present a literature review to clarify the reasons behind the importance of partnering with external actors for new product development and innovation, then we describe the CKN among the EE and its dealers´ network, and finally we discuss the organizational, technological and strategic dimensions of these interactions presenting some challenges facing the CKN and also especially some important factors that leads to success which is the creation, sharing and integration of knowledge in the new product development process of the extended enterprise.

The problems associated with passwords have led to the option of using biometrics as a secure authentication mechanism instead. In the case of passwords, you cannot guarantee that the person in possession of the password is the authentic owner. The inherent property of biometrics is that only the authentic owner can produce them. Biometric authentication does not require a user to remember anything, nor be burdened with the responsibility of creating a strong password [3, 4]. There is even a wide array of biometric types to choose from, such as fingerprint, iris, voice and facial geometry.

After considering the many advantages of biometrics, the disadvantages are just as plentiful. Should a biometric ever be compromised in anyway, it is not possible to change it, as you could a password. In addition, passwords can be kept secret, but a biometric is constantly being exposed to the outside world. With fingerprints, every time a person touches a surface, a latent fingerprint can be left behind. This fingerprint can be 'lifted' and be transformed into a latex copy [2]. An attacker with this latex copy can then masquerade as the authentic owner.

Furthermore, if one wanted to create an online biometric authentication server that could authenticate thousands of users simultaneously whilst making use of an adequate encryption scheme, the processing overhead should be considered. A biometric in electronic form is generally much larger than a conventional password. To transfer biometric data securely across the Internet using Public Key Encryption (PKE) requires substantial startup overhead for the client and the server, as well as significant processing abilities on the server. Without making use of PKE, BASE can also avoid the possible weaknesses associated with cryptographic protocols [5].

Although some people view biometrics as an intrusive technology, this paper is focused on the technological elements associated with secure remote biometric authentication, and not on people-related issues associated biometric technology.

This paper presents a prototype system that has been developed to securely authenticate clients based on their biometrics, across an insecure network (such as the Internet), without making use of encryption, passwords or timestamping techniques [6, 7]. This prototype system, known as BASE, is presented in this paper as an alternative solution to current biometric authentication systems. BASE solves the problem of lifted fingerprints, replay attacks, server spoofing and numerous other less significant threats.

2 Design

Although BASE is a prototype system, it has been developed to closely replicate the role it would have if it were being deployed in the real world. The potential applications for a secure remote biometric authentication system are endless, so BASE has been developed with a framework that can cater for multiple clients in a variety of situations.

2.1 Biometric Types

The primary biometric used in developing and testing this prototype system has been fingerprints. Fingerprints are among the most common form of biometrics being used and they are also one of the easiest biometrics to replicate. It was important to develop and test the BASE system with a biometric such as fingerprints to overcome the issue of 'lifted' fingerprints.

The BASE system is by no means limited to one specific biometric type. The core framework and protocol remains identical regardless of what type of biometric token is being authenticated. Only the matching algorithm used on the BASE server needs to be changed depending on the biometric being authenticated.

2.2 System Framework

The BASE system has been designed to allow for a variety of authentication scenarios without changing the framework. For this reason, the BASE server has been implemented as a Web Service. Web Services allow multiple clients operating on different platforms, to make use of the same service. A second advantage to using Web Services is that it can be used in most cases without adding additional rules to client or server firewalls. Web Services use SOAP, an XML-based message protocol. The transport protocols that are commonly used to transmit the XML-based messages are HTTP and HTTPS, both of which are not normally filtered by network firewalls.

The BASE server operates using a database, which manages each client's account. Each client has an associated account on the BASE server to store client information.

2.3 Potential Applications

Although BASE is not restricted to any specific environment, the architecture of the prototype is primarily built to provide authentication from remote locations. By this it is meant that the BASE client and BASE server communicate with each other over an insecure network, from separate physical locations.

This type of authentication can be particularly beneficial in electronic commerce environments. Clients could do their shopping online, and securely authenticate themselves using their biometric, after which they would be charged for their purchases. In these scenarios BASE can be used to enforce non-repudiation, since it can be guaranteed that the biometric is originating from the authentic owner. During an e-commerce transaction, the BASE authentication server will act as a transaction facilitator between a merchant, an electronic cash broker and a client.

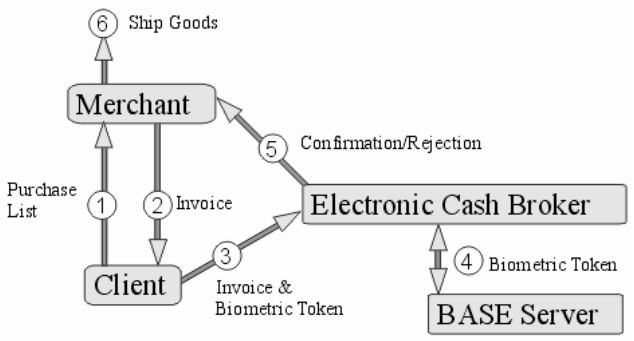

Fig. 1. E-commerce transaction using the BASE system.

The steps in Figure 1 above are as follows:

1. The client makes a selection of goods to purchase. The purchase list is sent to the merchant.
2. The merchant creates an invoice based on the client's purchase list. The merchant may chose to used encryption for transmitting the invoice, but this is not managed by the BASE system. The invoice is now sent to the client.
3. The client scans in his/her fingerprint, which is used by the BASE client application to create an authentication request (AR) message. The AR message is transmitted to the electronic cash broker, along with the invoice.
4. The electronic cash broker sends the AR message to the BASE server, to be authenticated. If the BASE server accepts the biometric, the electronic cash broker processes the invoice and makes the necessary cash transfers.
5. The merchant is now notified of the successful transaction.

In Figure 1, one should note that this scenario has been provided as a guideline for using BASE in an electronic commerce environment. It is possible to reorganize the entities involved in such an environment, so long as the biometric data generated by the BASE client, is transferred to the BASE server.

Because of the techniques used by the BASE system, if any illegitimate changes are made to the client's authentication request (AR) message at any stage, the BASE system will detect the change and reject the biometric.

2.4 Authentication Prerequisites

In order for clients to utilize the BASE system, they first need to be registered in the system. The registration process requires the client to physically travel to the institution making use of BASE technologies and scan their fingerprint. This scanned fingerprint is used as the registration template to authenticate the client during system operation.

After the registration template has been stored on the BASE server, the client is issued with a USB storage device. It is important to note that the content of the USB device should be inaccessible to anyone trying to access the data on it. A specific framework can be fused into the USB device to allow the device to act as a black box. An executable running on the USB device acts as the interface between the BASE client and the BASE server. The content on each USB storage device is specific to each client, and will be discussed in detail further in this paper.

The hardware protection of the USB storage device is beyond the scope of this paper. However, one recommendation is that a smart card can be used instead of a USB storage device, since smart cards have the ability to secure vital information located on the card.

3 Objectives

As previously mentioned, it was undesirable to make use of any encryption, passwords or timestamps during the transmission of biometric tokens. As a result the BASE system became vulnerable to a variety of attacks. Given that the intention of this prototype system was to introduce a different approach to secure authentication, these issues were solved without introducing encryption.

To facilitate secure biometric authentication, the BASE system identifies and prevents 4 main types of attack:

1. Simple replay attacks.
2. Tweaked token replay attacks.
3. Server spoofing attacks.
4. Lifted or copied biometrics.

3.1 Simple Replay Attacks

Every time the same client scans his/her fingerprint into a computer, it produces a different data stream. The chance of two fingerprint tokens from the same finger being exactly alike is almost impossible. Factors such as lighting, positioning and orientation of the finger, dirt and pressure can cause each scanned image to have subtle changes. These changes are reflected in the data stream transmitted to the server for authentication. The matching algorithm on the server is able to accept a biometric token provided it falls within an accepted threshold.

If an attacker intercepts an unprotected biometric token during transmission and replays the same biometric token at a later stage, it is easy to detect this simple replay attack. If the authentication server maintains an archive of previously accepted biometric tokens, and given the fact that no two biometric tokens should be the same, the authentication server can easily detect a simply replay attack if it receives a biometric token that has been sent before.

3.2 Tweaked Token Replay Attacks

In a password authentication system, the server only authenticates a user if the submitted password matches 100% to the stored original. This is known as symmetric matching. Biometric authentication works differently since no biometric token is the same. Biometric authentication uses asymmetric matching to determine if a biometric falls within an accepted threshold. This threshold will be configured on the server, when the BASE system is installed.

If an attacker intercepts a biometric token and tweaks the data slightly by changing a single byte and replays it, the server could accept it. The authentication server will see the submitted biometric token as a fresh token, since no record of it will exist in the archives. At this point the tweaked biometric token will be passed to the matching algorithm and could be authenticated, provided the token still falls within the accepted threshold.

3.3 Server Spoofing Attacks

PKE makes use of public/private key pairs that can be used to prove the authenticity of either the client or the server. Since BASE does not make use of PKE, it would be vulnerable to server spoofing attacks.

A client wishing to be authenticated could be deceived into sending a fresh biometric token to an attacker masquerading as the authentication server. The attacker could then send the 'stolen' biometric token to the authentic server to be authenticated. The authentication server would then illegitimately authenticate the attacker, since the biometric token being sent would not exist in the server archive and it would pass the matching algorithm check.

3.4 Lifted or Copied Biometrics

Fingerprints are particularly vulnerable to being lifted and have copies made from them. Currently it is very difficult or even impossible for an attacker to obtain other forms of biometric without the owner's cooperation, but in due time this may not be the case. The problem of lifted or copied biometrics is currently an important one, especially when using fingerprints for authentication, as in the BASE prototype system.

An attacker in possession of a copy of a client's fingerprint can easily be illegitimately authenticated. The authentication server would accept the submitted biometric token as a fresh token, and the token would also pass the matching algorithm check since it originated from the legitimate owner.

4 Implementation

In this section all the technical aspects of BASE are explained, as well as the communication that takes place between the client and server during authentication. As previously mentioned, a client needs to be registered in the BASE system before any authentication can take place. The registration process consists of three steps:

1. The client scans his/her fingerprint, producing a registration template. The registration template is stored on the server and used during authentication, for matching.
2. A Personal Token Archive (PTA) is created. A PTA is an archive consisting of space for 100 biometric tokens. During the registration process, the PTA is initialized to 100 randomly generated byte streams in equal length to a biometric token byte stream. The random byte streams in the PTA will slowly be replaced by biometric tokens during system operation. The reason for setting the size of the PTA to 100, is discussed in further detail in section 5.5.
   The PTA is copied to the client USB storage device and a copy of the PTA is stored on the BASE server. The associated token archive on the BASE server is referred to as the Master Token Archive (MTA).
3. A randomly generated key is created, referred to as the shared-key. The shared-key can be thought of as a serial number that is unique to each user. After the creation of the shared-key, it is copied to the client USB storage device and a copy is stored on the BASE server. The client never needs to be aware of the shared-key nor the PTA.

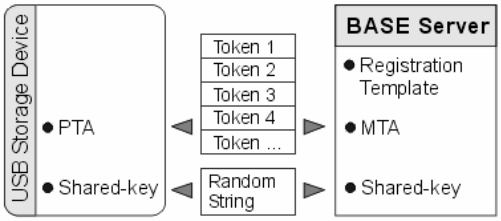

Fig. 2. Contents of the client USB storage device.

After the client has been registered in the system they are ready to make use of the BASE system for authentication. To be authenticated a client will need to insert their USB device into a machine, followed by scanning their fingerprint. The authentication process will follow.

4.1 Challenge Response Message

At the point where BASE clients need to be authenticated by the BASE server, they insert their USB storage device into a computer. A client executable located on the USB device executes, providing the client with an interface instructing him or her to place their fingerprint on the scanner. The biometric token produced from this scan is

referred to as the fresh biometric token. Once the client has produced the fresh biometric token, the following operations are handled by the client executable.

Firstly, the client executable requests a Challenge Response (CR) message from the BASE server. This can be thought of as a handshaking process. To request the CR message from the server, the client generates a random string and passes it as parameters to the web service method, GenerateChallengeToken, along with the client's user name.

```
Client >> [GenerateChallengeToken(user_name, random_string)] >> Server
```

Upon arrival of the CR message request, the BASE server randomly selects a token in the MTA (a random number between 1 and 100). This selected token is known as the Challenge Token.

The BASE server now computes three MD5 hashes:

- The hash of the index of the Challenge Token.
- The hash of the random_string received by the client.
- The hash of the shared-key.

All of the above three hashes are XOR-ed with each other to produce the CR Message. The CR message is then returned to the client.

```
Server >> [CR Message] >> Client
```

4.2 The Authentication Request Message

Once the client has received the CR message from the BASE server, the Challenge Token index needs to be extracted. To begin breaking down the CR message the client computes two hashes:

- The hash of the shared-key
- The hash of the random_string

These two hashes are XOR-ed with each other to produce the CR key. The CR key is then XOR-ed with the CR message. The resulting product of the XOR is the hash of the index of the Challenge Token. Finally, to determine what number the remaining hash represents, the client does a quick brute force attack with numbers 1 to 100 (since 100 is the size of the MTA and PTA). Once the client finds a match to the hashed number, the client is aware of the Challenge Token.

An Authentication Request (AR) Message now needs to be created by the client to send to the server. The AR message is used to authenticate the client and is made up of three parts:

A. The user name associated with the client.
B. The fresh biometric that was scanned in by the client is XOR-ed with the Challenge Token located in the PTA, requested by the BASE server.
C. The MD5 hash of the fresh biometric token is XOR-ed with the hash of the shared-key.

4.3 Authentication

The BASE server will receive the AR message from the client, and based on the legitimacy of the contents, the server can either authenticate or reject the client request.

Since the server is aware of what Challenge Token to expect, the fresh biometric can be extracted from part B of the AR message. This is done by XOR-ing the expected Challenge Token in the MTA with part B of the AR message.

Secondly, since the server is also aware of the shared-key, the hash of the fresh biometric can be extracted from part C of the AR message.

The fresh biometric from part B of the AR message is hashed and compared to the extracted hash contained in part C of the AR message. If the hashes don't match the client is rejected, if not, the server continues onto the last step.

The final step in authenticating the client would be passing the extracted fresh biometric token to the matching algorithm. Based on the results of the matching algorithm, the client is either accepted or reject. If the client is accepted, the BASE server replaces the Challenge Token in the MTA with the extracted fresh biometric token. The client is then notified of the outcome of the authentication in the Server Response (SR) message.

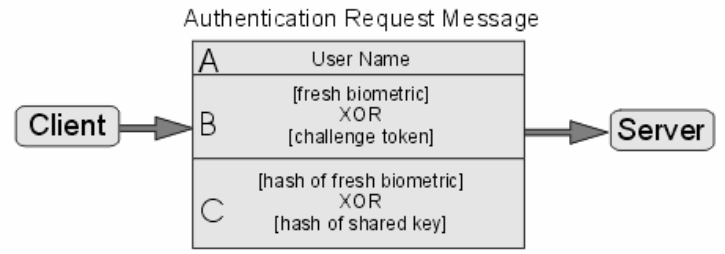

Fig. 3. The Authentication Request Message.

4.4 Server Response Message

The BASE server sends a clear text message indicating the success or failure of the authentication. No information needs to be protected in the SR message.

The client receives the server response message to determine whether or not the authentication was a success or not. If the client was authenticated successfully, then the Challenge Token in the PTA is replaced with the fresh biometric token that was used for authentication. After every successful authentication, the fresh biometric token replaces the Challenge Token in the PTA and MTA.

5 Evaluation

With the details of the BASE system explained, it will now be shown how each of the attacks previously identified are prevented by the BASE system. The four attacks identified were:

1. Simple replay attacks
2. Tweaked token replay attacks.
3. Server spoofing attacks.
4. Lifted or copied biometrics.
5. Hardware sniffing attack.

5.1 Simple Replay Attacks

Every time a client wishes to be authenticated, they first need to invoke the GenerateChallengeToken method on the BASE server. The server then returns the CR message containing the index of a specific Challenge Token to be used in the following AR message.

To initiate an authentication process, an attacker would first need to request a CR message. Since the attacker is not in possession of the shared-key, it is impossible to decipher the CR message. Should the attacker still attempt to replay the intercepted AR message, it will automatically be rejected. The BASE system never uses the same Challenge Token twice, so any AR message containing the wrong Challenge Token will cause the authentication to fail. The Challenge Token mechanism can be thought of as a one-time pad used during the construction of the AR message. The BASE system does not need to make use of timestamps because a replay attack will fail regardless of when the attack is attempted.

5.2 Tweaked Token Replay Attacks

In the case where an attacker attempts to tweak some part of an intercepted AR message and replay it, it will cause the BASE system to reject the AR message. The hash mechanisms contained within the AR message guarantee that the biometric data being sent has not been corrupted or tweaked in any way. The hash mechanisms used, inadvertently prevent AR messages with the wrong Challenge Token from being accepted.

5.3 Server Spoofing Attacks

The Challenge Response (CR) message requested from a client is used to prove the authenticity of the BASE server. An attacker spoofing the authentication server will not be able to produce a legitimate CR message because they are not in possession of the shared-key. If an attacker attempted to guess the shared-key, the CR message produced would make the Challenge Token index undeterminable by the client.

5.4 Lifted or Copied Biometrics

An attacker without the possession of a client's USB storage device would be unable to produce any legitimate AR messages. Every USB device contains a PTA and a shared-key, both of which are vital in deciphering the CR message and producing the appropriate AR message for authentication. Each USB device is specific to each client, so if an attacker used a lifted biometric with the wrong USB device, the system will reject the biometric token.

5.5 Hardware sniffing attack

During an authentication process, the client uses a fingerprint scanner to scan in their biometric. It is possible for an attacker to have port-sniffing software or hardware [8] to capture this fresh fingerprint data before it is even used by the BASE program. All fresh fingerprints are eventually placed in the PTA and MTA, and utilized in future authentication processes. Now, since the attacker has essentially obtained a PTA token (the sniffed fresh fingerprint from a previous authentication), they can construct an AR message by using a lifted fingerprint. The BASE system however, requires a random PTA token (the Challenge Token) to be used in each authentication process. For this type of attack to work, the attacker would need to use his/her sniffed token at exactly the right time. If an attacker submits his/her stolen token at the wrong time, BASE automatically detects this attack, and disables the client account. By increasing the size of the PTA and MTA from 100 to any larger number, reduces the chance of an attacker from using their stolen token at exactly the right time. Unfortunately, increasing the PTA and MTA size would also require more storage space on the server. The balance of security versus storage space must be decided on when setting up the BASE system.

6 Conclusion

The BASE prototype system has been developed and tested using the above attack scenarios. The system succeeds in preventing all of the attacks mentioned.

Due to the fact that the primary operation used by BASE is XOR, the system has extremely low processing overhead. Some cryptographic methods are used, but not PKE, thus the BASE system isn't affected by the problems associated with different encryption algorithms. The prototype system is also able to distinguish between false acceptance (legitimate rejections) and any form of attack. Since this is the case, BASE can stop multiple attack attempts after the first attempt without the need for an attempts counter [9].

If the BASE system were deployed for actual use, the client USB storage devices could be replaced by hardware protected smart cards or any device able to store and protect local data. The protection of the content on the client storage device is critical for the integrity of the system, but not considered apart of the scope of this paper.

Other biometric authentication systems exist [6, 10], however they make use of passwords, timestamps and public key cryptography to enforce security. The described BASE system may not be the only system that provides secure remote biometric authentication, but rather proposes an alternative solution with numerous benefits.

References

1. Halevi, S. and H. Krawczyk, Public-key cryptography and password protocols. ACM Transactions on Information and System Security (TISSEC), 1999. 2(3): p. 230-268.
2. Sukhai, N.B., Access control & biometrics. Information Security Curriculum Development, 2004: p. 124-127.
3. Tari, F., A.A. Ozok, and S.H. Holden, A comparison of perceived and real shoulder-surfing risks between alphanumeric and graphical passwords. ACM International Conference Proceeding Series, 2006. 149: p. 56-66.
4. Gupta, P., et al. Architectures for cryptography and security applications: Efficient fingerprint-based user authentication for embedded systems. in Proceedings of the 42nd annual conference on Design automation DAC '05. 2005.
5. Xu, S., G. Zhang, and H. Zhu, On the Properties of Cryptographic Protocols and the Weaknesses of BAN-like logics. ACM SIGOPS Operating Systems Review, 1997. 31(4): p. 12-23.
6. Khan, M.K. and J. Zhang, Improving the security of 'a flexible biometrics remote user authentication scheme'. Computer Standards & Interfaces, 2007. 29(1): p. 82-85.
7. Kim, H.S., S.W. Lee, and K.Y. Yoo, ID-based password authentication scheme using smart cards and fingerprints. ACM SIGOPS Operating Systems Review, 2003. 37(4): p. 32-41.
8. Roberts, C., Biometric attack vectors and defences. Computers & Security, 2007. 26(1): p. 14-25.
9. Ding, Y. and P. Horster, Undetectable on-line password guessing attacks. ACM SIGOPS Operating Systems Review, 1995. 29(4): p. 77-86.
10. Lin, C.-H. and Y.-Y. Lai, A flexible biometrics remote user authentication scheme. Computer Standards & Interfaces, 2004. 27(1): p. 19-23.