Modified Private Key Generation for Biometric Signatures

PAWAN K. JANBANDHU, M. Y. SIYAL
School of Electrical and Electronic Engineering
Nanyang Technological University,
Information Systems Research Laboratory,
Singapore 639798

Abstract :- Although PKI offers many advantages like authentication, integrity, non-repudiation and confidentiality, it cannot identify the maker of a transaction. It can only identify maker’s belongings (disks, smart cards) or what he remembers (passwords, PINs etc). Integrating biometrics with PKI to directly generate private key can not only identify the maker accurately but can also resolve the key management issue by not having to store the private key anywhere. We proposed a new system to generate digital signatures using biometrics and denominated it as Biometric Signatures in [1,2]. We also discussed two schemes to generate Biometric Signatures using RSA and DSA. This paper gives detail description of the modifications suggested for private key generation in [1,2] using both schemes to allow certificate renewal. This paper also proposes a new method to generate private keys for Biometric Signatures using RSA. Speed of Biometric Signatures using modified schemes for iris recognition and comparative key generation speeds for various biometrics is presented using JAVA implementation of both approaches.


1 Introduction

Public Key Infrastructure (PKI) offers authentication, message integrity, confidentiality and non-repudiation. However it cannot assure identity of the maker of a transaction; it can only identify the maker’s belongings (computers, disks, smart cards) or what he remembers (passwords, PINs etc). The private key is stored in hard disks protected by 6 to 8 character passwords or carried in smart cards/floppy disks etc. However, passwords/PINs could be cracked by guessing and other means and smart cards/floppy disks could be lost or stolen. Thus, an imposter can easily masquerade as a legitimate user and defraud the system. Researchers have invented a new mechanism to minimize the risk for security breach of private key: distributed generation of RSA keys over more than one server, thereby, dividing the key in shares for each server [13][14]. This will not only increase the effort required by hackers to steal the private key and therefore discourage them from even attempting but can also be used to delegate the authority of signing documents to more than one person in the company to minimize errors or misuse from people within the company.

Biometric-PKI combination is another solution being explored by researchers and developers for user authentication for e-commerce security. A biometric is a person’s unique physical or behavioral characteristic that can be used to identify the individual. Physical characteristics include fingerprints, hand or palm geometry, retina, iris and facial characteristics. Behavioral characteristics include signature, voice, keystroke pattern and gait [12][22]. Due to uniqueness, biometric is the only way to identify a person with sufficient legal background. Biometrics are being used in many applications like physical access control, national ID database to confirm identity, ticketless travel, commuting and maintaining health records, online banking via internet and ATMs, secure computer log-on, website access, password file access etc. Fingerprint technology is the most widely used biometric [18][20][22]. Iris recognition [4][5][8][16] is a highly accurate biometric based recognition technology. With John Daugman’s iris recognition algorithm, one can achieve Equal Error Rate (ERR) of as low as 1 in 1.2 million [16].

One of the solutions suggested to resolve key management issue is to use biometrics for private key access. Using biometrics to protect private keys requires tighter integration of biometrics with the operating system [3] to prevent attacks from hackers.
Many companies have developed biometric based products and their own standards for user identification and key generation like “Bioscrypt” by Toronto based Mytec Technologies Inc. [15][17]. The notion of using biometric template directly as a cryptographic key was first proposed by Bodo in a German Patent. Extending this approach we proposed a new biometric based signature system in which biometric was integrated with PKI to generate the signature keys and denominated it as “Biometric Signatures” in [1,2]. The system is secure, efficacious, faster, convenient, non-invasive and correctly identifies the maker of a transaction. Although we discussed Biometric Signatures and its implementation using DSA and RSA in [1] and [2] in detail we briefly describe Biometric Signature, its advantages and its implementation using RSA and DSA in this paper in Sec. 2 for the sake of clarity followed by the modifications required in the two schemes proposed in [1,2] to enable key renewal in detail and a new method to generate signature key of desired length from any biometric of any template size for Biometric Signatures using RSA algorithm. We also give the speeds of the modified schemes for iris recognition and comparative key generation speeds for various biometrics using JAVA implementation.

2 Biometric Signatures
Biometric Signature is integration of biometrics with PKI for digital signatures by generating the signature key from a stable biometric template with or without modification. The main advantages of this method for digital signing are listed below:

1. This method will correctly identify an individual and not a person’s belonging or what he remembers.
2. No storage of biometric template required to retrieve the private key (since, they can be regenerated on demand). Therefore, eliminates problem of vulnerability of stored private keys for PKI (resolves key management issue). Even owner doesn’t know what is his private key.
3. Offers all advantages of PKI and digital certificates (authentication, integrity, confidentiality, non-repudiation).
4. No transmission of templates over internet.
5. Provides more convenience in signing documents. One can sign documents anytime anywhere using pinhole cameras implanted into their PDA’s, laptops, cell phones etc. based on non-invasive biometric like Iris.

6. Biometric Signature can also be used for user authorization with minor modifications e.g. by encrypting a randomly generated or previously known small message (send by server on demand) and transmitting it back to the server for verification using client’s public key. No, storage of template database is required on server side hence, no template misuse.

This approach requires all the bits of the biometric template to be “correct” otherwise verification would be impossible. The veracity of the private key generated can be checked locally before actual signature by verifying a small message signed using the private key generated.

Although Biometric Signature can be implemented with any stable biometric, we suggested DNA in [1] and [2]. Other most promising biometric is iris recognition due to its high accuracy (ERR of 1 in 1.2 million), long-term stability, ease of use and non-invasiveness.

2.1 Biometric Signature using RSA Algorithm
RSA algorithm can be used with 512 byte iris template to generate Biometric Signature in the following manner. The length of the template can be brought down to 128 bytes or closer using some irreversible or one way hash function (similar to hash functions with larger bit output) or a combination of functions that will generate a unique biometric template representation of desired length and use it to generate the decryption exponent, d. One such approach suggested in [1,2] was to feed the iris template to MD5/SHA1 to generate an AES key, and then use that in counter mode to generate as many bits as needed. We now propose a new method to generate the private key, which is to feed biometric template to HMAC-SHA1/MD5 with different keys to compute different MACs, concatenate them to get the hash of desired length and use it to generate d. Using HMAC-SHA1 with six different keys one can obtain 120 byte hash. Similarly, with HMAC-MD5, with 8 different keys one can obtain exactly 128 byte hash. Since, decryption exponent, d should roughly be of the same size as the modulus length for extra security [11], we choose length of hash to be 120/240 bytes. Choose p and q to be 64 byte numbers so that size of modulus n = p*q = 128 bytes. Compute Euler Totient function, \( \phi(n) = (p-1)(q-1) \) and decryption key, d from 120 byte hash obtained from HMAC-SHA1 by incrementing it to get a closest number relatively prime with \( \phi(n) \). Private key = \((n,d)\). Compute encryption exponent, e as the multiplicative inverse of d modulo \( \phi(n) \) using:
\[ e = d^{-1} \bmod (\phi(n)) \quad [Public \ Key = (n,e)] \]

Rest is same as the digital signature scheme using RSA. Message digest can be generated using MD5/SHA1. Also see [1][2] and [6] for more details. Since private key is never transmitted, so there is no question of misuse of biometric template by the receiver end. However, using this method, one needs to store \( \phi(n) \) instead of the decryption exponent, \( d \). Private key will be generated on presenting live biometric before an image acquisition camera and combining it with Euler Totient function as mentioned above.

2.2 Biometric Signature using Digital Signature Algorithm (DSA)

Digital Signature Algorithm, DSA was proposed by U.S. National Institute of Standards and Technology (NIST) in 1991 for use with Digital Signature Standard (DSS). It is a variant of the Schnorr and ElGamal signature algorithms [10][11]. Biometric Signature using DSA can be achieved by generating the private key by computing 160 bit hash value of biometric template of any size using one way hash function SHA1 and assigning it to \( x \). Rest is same as proposed by NIST in reference [7].

2.3 Security of Biometric Signatures

In the first scheme, RSA offers maximum security due to the huge key size involved (approx. 128/256 bytes). If the Euler totient function is ever compromised i.e. if the security of keystore (directory where CA certificates, \( \phi(n) \), etc. are stored) is ever breached or smart cards containing \( \phi(n) \) keys are lost or stolen, the attacker can obtain the decryption exponent by computing multiplicative inverse of \( e \) modulus \( \phi(n) \) [6]. Thus, security of this method is no better than that provided by storing the private key directly. If attacker managed to obtain \( \phi(n) \), he can also compute the biometric template by reverse engineering (finding the closest number that matches the required criteria as mentioned in section 2.1 for key generation).

Biometric signature using second approach is obtained at no compromise with the security or speed of DSA. In fact, it increases the security of the digital signature algorithm by not having to store the private key \( x \), on hard disks.smart cards etc. It can be generated directly on demand by presenting a live iris before a camera. Security of \( k \) remains the same.

2.4 Certificate Renewal, Modified Private Key Generation and Storage Requirement

Digital certificates are valid for a give period. When the certificate expires, it is revoked automatically by the issuing certification authority, CA. One can also revoke the certificate if the private key has been compromised. In either case, one can get a new certificate by computing a new pair of signature keys. In case of RSA this can be achieved by computing a new pair of \( p \), \( q \) and Euler totient function. However, since, the attacker can easily compute the biometric template from \( \phi(n) \) as explained above, he can find the new private key by finding the closest number to the template that corresponds to the new public key for encryption and decryption of a known message by simply incrementing it, thereby, forfeiting the use of biometric template forever. In case of biometric signature scheme using DSA, if the 160 bit key is compromised, one cannot retrieve the biometric template (depends on security of SHA1) but the use of the scheme to generate private keys will be forfeited since there is no way to generate a new key.

This can be resolved by modifying the template with another function with random output like XOR the template with a randomly generated number, \( R \) and then feed the resulting number to the hash function to produce a new 160 bit private key as shown in Fig. 1. One needs to preserve that number for future signatures. Since output of XOR function is going to a hash function (SHA1) with 160 bit output, \( R \) could be as small as one byte number. It could be chosen to be owner’s favorite number for easy remembrance. This will also avoid storage of \( R \). \( R \) can also be integrated within the application to regenerate the private key for subsequent digital signatures. Attacker cannot generate the private key from \( R \) alone even if he managed to steal it from the application. Hence, if the private key is compromised one can still obtain a new private key with no compromise in existing security by changing the \( R \) value.

Similar thing can be done for biometric signature using RSA as shown in Fig. 2. However, in this case when HMAC-SHA1/MD5 is used to generate desired size hash from a biometric template, no \( R \) is required since, a new pair of signature keys can be generated by changing anyone/all of the keys used to generate the hash as explained in sec. 2.1. \( R \) is only required when SHA1-AES is used to generate the hash also explained in sec. 2.1.
Hence, the modified key generation process for the two schemes discussed for biometric signature with RSA and DSA allows certificate renewal on demand without forfeiting the use of the biometric for future signatures.

Biometric Signature using RSA requires storage of $O(n)$, n and R/HMAC keys for signature generation and public key $(n,e)$ for verification. DSA requires storage of R only. Storage of R can be avoided by keeping it as small as possible (for easy remembrance).

3 Implementation of Biometric Signatures

Biometric Signatures using RSA and DSA with modified private key generation as shown in Figures 1 and 2 were implemented in Java. The average time taken by both algorithms during various stages is tabulated in tables 1, 2 and 3. No secret key encryption was done to transmit message and signature to the receiver for simplicity. All computations were carried out using Windows 98, Intel PIII 500 MHz, Sun Java SDK 1.3.0 platform. Message used was chosen to be a text file of 622 bytes. R was chosen to be 25.

3.1 Discussion of Experimental Results

Table 1 illustrates the time taken for modified scheme for biometric signature using RSA for iris recognition (template size = 512 bytes). HMAC-SHA1 was used as the one-way function to map the template to 120 and 240 bytes of hash by concatenating the result obtained using 6 and 12 different keys for modulus lengths of 1024 and 2048 bits respectively. From the table, we can see that the key generation (computing n, d and e) time is increased for modulus of 256 bytes due to larger key length involved. Signature and verification time are considerably small.

Table 2 illustrates computation time for modified scheme for biometric signature using DSA for 512 byte iris template. We can see that the precomputation time (time to compute p, q and g) is huge compared to the key generation $(x,y)$, signature $(r,s)$ and verification time. This is due to the lengthy procedure (recommended by NIST in [7]) employed to compute p and q. Also, p and q need not be generated again for each signature and can be shared among a group of users.

Table 3 shows comparative speeds for signature key generation for various biometrics. As the template size [22] of the biometrics increases, time taken for key generation also increases as expected. Greater speeds can be achieved with C and hardware implementation of the schemes.

Thus, Biometric Signature using both RSA and DSA can be implemented with any biometric with no restriction on template size. Also, since verification time is very less, veracity of private key generated can be checked locally in very short time (in fraction of a seconds) as explained in sec. 2.
3.2 Problems in Implementing Biometric Signatures for Practical Purposes

The main problem in integrating biometrics with PKI for Biometric Signatures is the requirement for all bits to be “correct” which is difficult to achieve. Accuracy of any biometric recognition technology is measured in terms of False Accept Rate (FAR), False Reject Rate (FRR) and Equal/Cross-over Error Rate [4][9][19][22]. For most biometrics FAR is usually prespecified at <0.0001 [20]. Exceptionally, iris recognition can give ERR of 1 in 1.2 million. However, according to John Daugman (inventor of iris recognition) about 10 to 15 percent of the bits change with each presentation, just because of the difficulties of obtaining a good image of the iris at a distance of about a meter, and sometimes through contact lenses, with eye lashes, eyelid occlusion, camera noise, reflections from the cornea, etc. Error correction methods do not work, because such a large number of bits could be “incorrect,” and many of them could be clustered in one place [9][21].

DNA is the best biometric for Biometric Signatures which is known to be unique with a well defined set of features of ones and zeroes. With the advancements in technology (increase in resolution power of image capturing devices like CCD monochrome camera, improvement in biometric recognition algorithms etc. and human precaution during image acquisition, Biometric Signatures could be possible in near future with other biometrics.

4 Conclusions

Biometric Signatures, its advantages and two schemes to generate biometric signatures using RSA and DSA are discussed in brief. Biometric Signature is a new approach to integrate biometrics with PKI to digitally sign a document using biometric based digital signature key generation that is secure, faster, convenient and accurately identifies maker of a transaction. It does not replace digital signatures completely but is only used to generate the private key. However, Biometric Signatures resolves the key management issue in PKI by avoiding storage of private keys or biometric templates anywhere. It can be generated on demand by presenting a live biometric before an image capturing device. It combines the advantages of PKI (integrity, authentication, confidentiality, and non-repudiation) and biometrics (personal identification).

Modifications suggested for private key generation in [1,2] using RSA and DSA to allow certificate renewal are discussed in detail. Use of HMAC-SHA1 as one-way function to map biometrics of any template size to desired length for generating private keys for Biometric Signatures using RSA is proposed and implemented. Speed of Biometric Signatures using modified schemes for iris

---

Table 1 Biometric Signature using RSA speeds for different modulus lengths for template size of 512 bytes. HMAC-SHA1 was as used to generate the 120 and 240 byte hash.

<table>
<thead>
<tr>
<th></th>
<th>1024 bits (secs)</th>
<th>2048 bits (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key Generation</td>
<td>3.65</td>
<td>19.28</td>
</tr>
<tr>
<td>Signature S, (in sec)</td>
<td>0.27</td>
<td>1.53</td>
</tr>
<tr>
<td>Verification (in sec)</td>
<td>0.22</td>
<td>1.54</td>
</tr>
</tbody>
</table>

Table 2 Biometric Signature using DSA speeds for different modulus lengths with a 160 bit Exponent.

<table>
<thead>
<tr>
<th></th>
<th>512 bits (secs)</th>
<th>768 bits (secs)</th>
<th>1024 bits (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generation of p, q, g</td>
<td>17.01</td>
<td>45.01</td>
<td>120.30</td>
</tr>
<tr>
<td>Generation of Keys (x, y)</td>
<td>0.24</td>
<td>0.24</td>
<td>0.24</td>
</tr>
<tr>
<td>Signature (r,s)</td>
<td>0.20</td>
<td>0.22</td>
<td>0.27</td>
</tr>
<tr>
<td>Verification</td>
<td>0.06</td>
<td>0.06</td>
<td>0.11</td>
</tr>
</tbody>
</table>

Table 3 Comparative Private Key generation speeds for Biometric Signatures for various biometrics for modulus of 1024 bits.

<table>
<thead>
<tr>
<th></th>
<th>Template Size (in bytes)</th>
<th>RSA (with HMAC-SHA1) Generation of n, d, e. (in sec)</th>
<th>DSA Generation of x, y (in sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hand Geometry</td>
<td>9</td>
<td>1.63</td>
<td>0.11</td>
</tr>
<tr>
<td>Retina</td>
<td>96</td>
<td>1.92</td>
<td>0.14</td>
</tr>
<tr>
<td>Finger Scan</td>
<td>250</td>
<td>2.22</td>
<td>0.18</td>
</tr>
<tr>
<td>Iris</td>
<td>512</td>
<td>3.65</td>
<td>0.24</td>
</tr>
<tr>
<td>Face</td>
<td>1300</td>
<td>4.52</td>
<td>0.29</td>
</tr>
<tr>
<td>Signature</td>
<td>1500</td>
<td>4.74</td>
<td>0.36</td>
</tr>
<tr>
<td>Voice</td>
<td>7000</td>
<td>14.37</td>
<td>1.64</td>
</tr>
</tbody>
</table>
recognition and comparative key generation speeds for various biometrics is presented using JAVA implementation of both approaches (RSA and DSA).

Biometric Signatures can be implemented using any biometric which can guarantee long-term stability and high accuracy without restriction on its template size. Private key can be conveniently renewed periodically (say every 6 months to one year depending on the stability of the biometric) or on demand (e.g. in case of security breach). Thus, Biometric Signature system using both RSA and DSA facilitates certificate renewal without forfeiting the use of biometric for future signatures. DNA is the best biometric to be integrated with digital signature algorithms for Biometric Signatures.

Acknowledgement:
We are thankful to Ronald Rivest and John Daugman for helpful comments and suggestions.

References: