Design and Analysis of Methods for Signing Electronic Documents Using Mobile Phones

Pramote Kuacharoen
School of Applied Statistics
National Institute of Development Administration
118 Serithai Rd. Bangkapi, Bangkok 10240 Thailand
pramote@as.nida.ac.th

***Abstract*—Mobile phones have become essential for modern living. They have influenced the way we live. Besides using a mobile phone to communicate, we use it to take pictures, listen to music, watch video, play games, browse the Internet, and much more. Mobile phones continue to play an ever increasing role in our lives, so it is imperative to use them to enhance the convenience of our everyday lives. Estimates have mobile phone penetration expected to reach a vast majority of the population. Therefore, mobile phones should in the future be used to sign electronic documents.**

This paper presents three methods for signing electronic documents using mobile phones. The first method uses SMS to communicate between the server and the mobile phone, i.e., the hash value of the document is sent from the server via SMS, signed at the mobile phone, and sent back to the server via SMS. The second method uses QR codes where the hash value of the document is generated and displayed as a QR code at the client machine. The QR code is then scanned using a mobile phone. Subsequently, the digital signature of the document is created and transmitted to the server. The third method uses direct download where the document is downloaded to the mobile phone. The approved document can be signed and the digital signature is transmitted to the server. The security of each method is also analyzed and compared to the traditional method where a cryptographic token is used.

Keywords-digital signature; electronic document; mobile phone; non-repudiation; security

I. INTRODUCTION

The vision of paperless office has been discussed since 1970's, but over 30 years it is not yet a reality for most organizations [1]. It is not that the technology has not kept up with the vision. Rather, the reason that the paperless office does not exist lies with end users and the hardware. Some suggest that a less paper-centric environment may serve as a more realistic goal [2][3]. People are not comfortable using the technology or it is inconvenient to use the technology. The authenticity of the documents also becomes a concern. The paper-based documents can ensure their validity by having the responsible party sign the papers. However, as more and more organizations are transforming into a paperless office, traditional ink signing becomes inapplicable. The technology for authenticating electronic documents is readily available. A digital signature can be used to authenticate the signer of a document and ensure that the content of the document has not been altered.

A Document Management System (DMS) is an important part of a paperless office. The system allows electronic documents to be stored so that they can be retrieved as needed. Some document management systems have a built-in workflow module. The flow of the document through an organization follows a certain rule. Some documents must be approved by the superior or pass through an approval process. Traditionally, user ID and password are used to log into the system and the approval process is merely controlled by programming logic. Therefore, this system is vulnerable to forgery and tampering attacks. The documents can be modified after they have been approved. To improve security, a digital signature scheme can be used since it provides non-repudiation.

With the use of cryptographic tokens, the user can digitally sign documents by producing digital signatures. However, the user has to carry an extra device. Moreover, the device can only be used with a computer on which the driver for a particular device was installed. This may be inconvenient for some people. On the other hand, many people usually carry a mobile phone with them; therefore, mobile phones should be used to sign electronic documents.

The number of mobile phone subscribers has been steadily increasing in the past decade. International Telecommunication Union (ITU) estimated that the mobile phone penetration has reached 67 per 100 inhabitants globally by the end of 2009 [4]. In developed countries, mobile phone subscriptions have surpassed the population. Many people always have their mobile phones ready to use. Mobile phones are increasingly becoming important devices for our lives. People will depend on their mobile phones in daily activities. The modern mobile phone is capable of doing nearly everything in comparison to a desktop computer, but with the potential for more meaningful relevance and convenience to our daily activities.

In this paper, the design and analysis of three methods for signing electronic documents using mobile phones are presented. Each method can be adopted depending on the mobile infrastructure of a specific location where the technology is available. Some areas may have 2G mobile infrastructure while others may have 3G. With the use of mobile phones to sign documents, it may help us moving toward a paperless office.

This paper consists of six sections. The next section, Section II, provides background information and related work. Section III presents electronic document signing methods. Section IV analyzes the security of each approach.

Section V describes the implementation of each method. The last section, Section VI, concludes the paper.

II. Background and Related Work

This section provides background information used in this paper such as digital signatures, cryptographic tokens, QR codes, and Short Message Service.

A. Digital Signatures

A digital signature is a bit pattern that depends on the message being signed and uses some information unique to the signer. A simplified digital signature generation process is shown in Figure 1. The message M is fed into a cryptographic hash function resulting in a hash value h or a message digest. The hash value h which depends on the message M is encrypted using the signer's private key producing the signature.

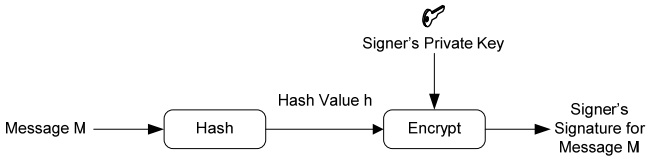

Figure 1. A simplified digital signature generation process.

To verify whether or not the digital signature is valid, the resulting hash value from the Message M' is compared to the value from decrypting the signature using the signer's public key. If both values are identical, the owner of the public key is the author of the message. Otherwise, the signature is invalid. A simplified digital signature verification process is illustrated in Figure 2.

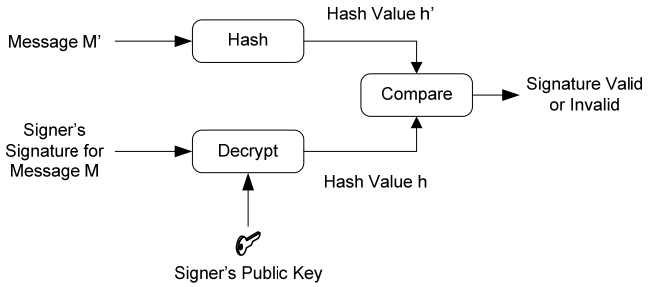

Figure 2. A simplified digital signature verification process.

Digital Signature Standard (DSS) [5] includes three techniques, namely; the Digital Signature Algorithm (DSA), the RSA digital signature algorithm [6], and the Elliptic Curve Digital Signature Algorithm (ECDSA) [7]. The security of the digital signature depends on the cryptographic hash function and the public key cryptographic algorithm. For breaking a digital signature, an attacker may create a fraudulent digital signature by creating a new message for an existing digital signature which is an attack on the cryptographic hash function or by constructing a fraudulent digital signature for a given message which is an attack on the public key cryptographic algorithm. The hash function must be collision resistant and the public key algorithm must be strong against attacks. The approved techniques are considered secure.

It is computationally infeasible to forge a digital signature. The digital signature provides authentication and non-repudiation. Therefore, if the signature is valid, the author of the message cannot deny creating the message.

B. Cryptographic Tokens

A cryptographic token is a physical device that provides the ease of authentication. The cryptographic token may store passwords, public-keys, and private keys and may be capable of performing computations such as encryption, decryption, generating digital signatures, and verifying digital signatures. For example, USB cryptographic tokens allow the user to generate digital signatures. RSA Laboratories defines cryptographic token interface standard [8] which specifies an application programming interface (API), called "Cryptoki" to devices which store cryptographic information and perform cryptographic computations.

C. QR Codes

A QR code is two-dimensional code which consists of a black square pattern on white background. The QR code contains information in the vertical direction as well as the horizontal direction. The data capacity can be the maximum of 7,089 numeric characters, 4,296 alphanumeric characters, or 2,953 bytes [9]. QR codes use the Reed-Solomon error correction which can detect and correct multiple errors. QR codes can be read by QR scanners or mobile phones with a camera.

The snapshots of QR codes taken by mobiles phones usually are not perfectly aligned causing the image to be distorted. However, algorithms for correcting distorted images exist. Ohbuchi et al. [10] show new algorithms and implementations for reorganizing QR codes in mobile phones. Sun et al. [11] present an algorithm for analyzing and correcting the distorted QR code image.

Dodson et al. [12] propose a challenge-response authentication system for web application using QR codes. The QR code contains a challenge and a link to the server. On the login page, a QR code is displayed for the user to take a picture with a mobile phone. The response is then sent to the server from the mobile phone. Upon successful authentication, the web page is refreshed allowing the user to login.

D. Short Message Service (SMS)

Short Message Service or SMS is one of the most widely available and popular mobile phone users. SMS enables sending short messages up to 160 ASCII characters or 140 bytes between mobile phone users. An SMS message is sent in store-and-forward fashion. It may pass through an unsecure link such as the Internet. The message may be eavesdropped or modified during transit. Moreover, there are threats associated with SMS spoofing where the originating mobile number is replaced with alphanumeric text. There are proposals for enhancing the security of SMS. Hossain et al. [13] propose a security scheme for improving

the SMS security. Wang [14] presents an anti-counterfeiting system based on SMS.

III. Methods for Signing Electronic Documents

In this section, the design of three methods for signing electronic documents in insecure environments using mobile phones is presented. The mobile phone stores the private key of the user. The private key is protected using a passphrase.

A. SMS Method

For signing documents using SMS, a simple method will be explained following by a security-enhanced method. In the simple method, the system components consist of a user, a mobile phone, a web browser on a PC, and a DMS. The user uses the web browser on a PC to view documents. When the user selects a document to view, the web browser requests the document from the DMS. The DMS responds with the requested document. The web browser displays the document. After reading the document, the user may decide to sign the document by pressing a button. The web browser forwards the request to the DMS. The DMS then calculates the hash code of the document and sends an SMS message containing the resulting hash code to the user. After receiving the SMS message, the user may choose to sign the document. The application on the mobile phone creates the digital signature of the document using the user's private key and sends the digital signature to the DMS via SMS. The DMS verifies the signature and sends an SMS confirmation. A sequence diagram for signing an electronic document using SMS is illustrated in Figure 3.

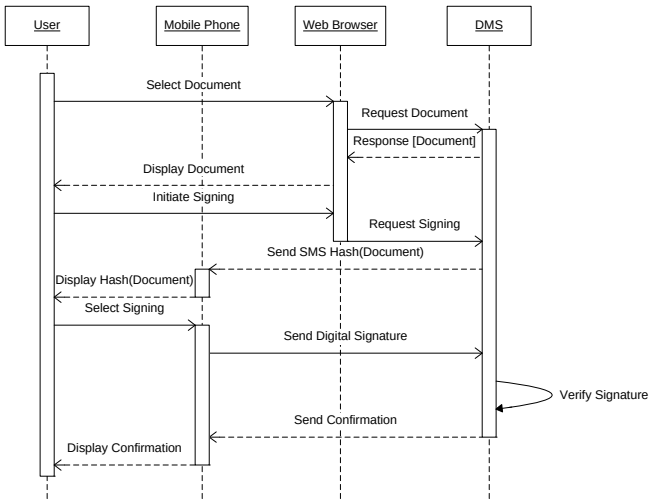

Figure 3. A sequence diagram of a simple method for signing an electronic document using SMS.

In this simple method, it is obvious that the user must trust the DMS since the user cannot be certain that the received hash code corresponds to the message to be signed. Moreover, the SMS message is sent through an insecure mobile network. The message can be subjected to modification during transit.

To ensure that the user signs the intended document, the user must be able to compare the received hash code and the hash code of the document. Figure 4 shows a security-enhanced method which is similar to the simple method. When the user selects a document to view, the document is downloaded to the web browser. The embedded software on the browser computes the hash code of the document and displays it on the browser. The user can visually compare the calculated hash value to the received one before signing. If they are matched, the user can be sure that the correct document is to be signed.

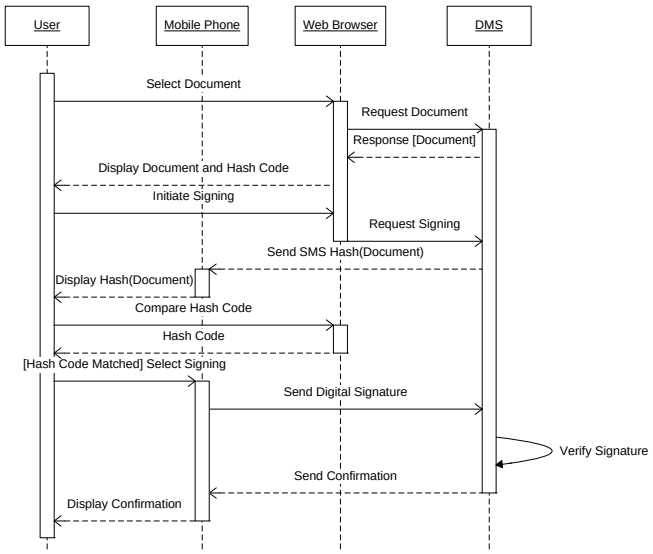

Figure 4. A sequence diagram of a security-enhanced method for signing an electronic document using SMS

Since the SMS limits the message size to 140 bytes, it may require sending multiple messages for a system that requires a strong encryption algorithm. If the technology is available, an MMS message can be used to serve the purpose.

B. QR Code Method

Similar to the previous method, the QR code method involves a user, a mobile phone, web browser on a PC, and a DMS. However, after the web browser receives the document, the embedded program computes the hash value of the document and generates a QR code of the hash value accordingly. After reading the document, the user may approve the document by signing it. The process of signing can be accomplished by using a mobile phone with a camera to take the picture of the QR code. The application on the mobile phone converts the QR code into data which is the hash value of the document. The digital signature can then be produced. The user confirms the signing process and the application sends the corresponding digital signature to the DMS. Figure 5 shows a sequence diagram for signing an electronic document using QR codes.

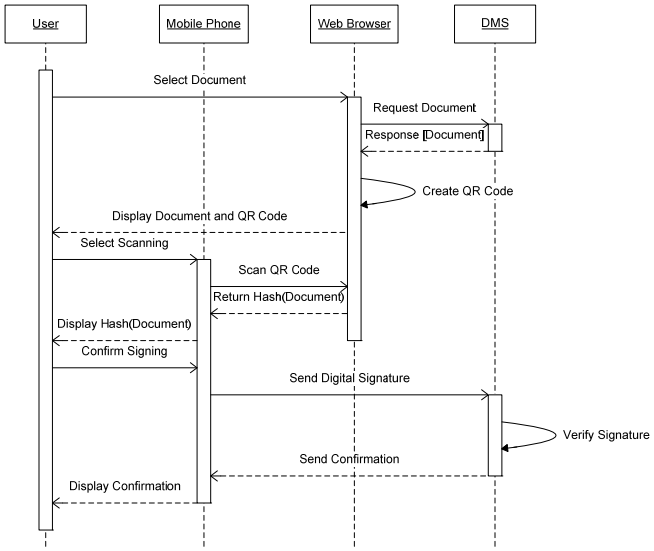

Figure 5. A sequence diagram for signing an electronic document using QR codes.

C. Download Method

In contrast to the previous two methods, the download method does not involve the web browser on the PC. For this method, as shown in Figure 6, the user downloads a document to the mobile phone and views it. When the user approves the document, the application on the mobile phone computes the hash value and creates the digital signature using the user's private key. The application then sends the digital signature to the DMS. Upon successful verification of the digital signature, the DMS sends a confirmation to the user.

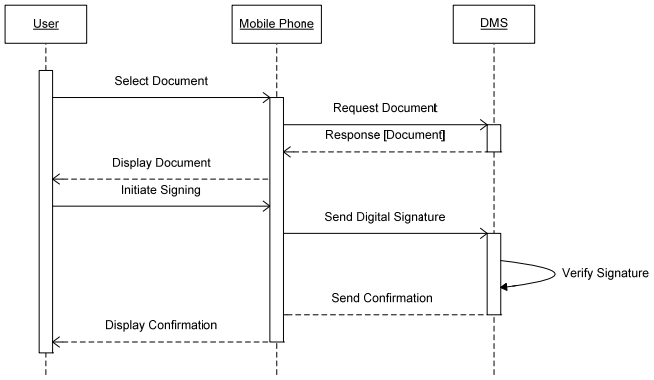

Figure 6. A sequence diagram for signing an electronic document using download.

IV. SECURITY ANALYSIS

This section describes the security analysis of signing electronic documents using mobile phones in comparison with using cryptographic tokens. To use a cryptographic token, the user typically inserts the device into a USB port. After plugging the token into the port, the user enables the device by entering a password or a passphrase. When the user wishes to sign a document, the token reads the document and produces a digital signature using the user's private key stored on the token. A compromised DMS cannot fool the user to sign an unintended document whereas a compromised device driver can send the token a different document from the one which the user sees on the screen. However, this model of signing electronic documents is considered to provide strong security.

For signing a document via SMS, the DMS computes the hash value of the document and sends it to the user. A compromised DMS can send a hash value of another document to the user. The SMS infrastructure is not secure. Therefore, the SMS message can also be modified during transit. However, this approach only requires a mobile phone which can run an application to encrypt data. If the security level requirement is low to medium, this approach is still attractive. The security of this method can be enhanced by trading off the user's convenience. The user must do more work in order to be certain that the intended document is being signed. In this security-enhanced method, the software on the browser computes the hash value of the document and displays it on the screen. When the user receives the SMS, the displayed hash value and the received hash value can be visually compared before signing. This prevents the user from signing some unintended documents or a modified hash code. However, the embedded software on the browser must be signed to ensure its integrity. Therefore, this method can provide strong security.

When a QR code is used, the hash value generated at the client machine, converted to a QR code and scanned using a camera. The application on the mobile phone converts the image of the QR code back to the hash value of the document and creates the digital signature of the document before sending it to the server. The embedded software on the browser must also be signed. Therefore, its integrity is guaranteed. Attacking on this scheme requires compromising the mobile phone. The security of this method of signing documents is comparable to using a cryptographic token. This approach requires a mobile phone with a camera that can take a picture of the QR code.

In the last method, the document is downloaded to the mobile phone. The user can view and sign the document. The attacker cannot fool the user to sign a different document, unless the mobile phone is compromised. The threat from an attacker fooling the user to sign something is low. Therefore, this method is also considered secure. This approach requires a high bandwidth data transmission and the mobile phone must be capable of displaying documents.

V. IMPLEMENTATIONS

Each method is verified using Java Cryptographic Architecture (JCA) and Android. SHA-256 and RSA are used as the algorithms to generate digital signatures.

For the SMS method, the server sends a document number and the hash value of the document to the user. The length of the document number is four bytes and the length of the hash value is 32 bytes (256 bits). The server message fits in one SMS message. The user sends a message consisting of the received document number, the certificate

number, and the signature to the server. The length of the certificate number is four bytes. Since a 1024 bit RSA key is used to sign the document, the length of the signature is 128 bytes. As a result, the length of the message is 136 bytes which can be contained in one SMS message. Figure 7 shows the formats of messages.

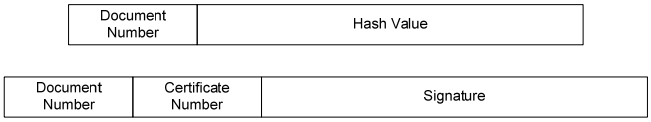

Figure 7. Message formats.

For the QR code method, the document number concatenated with the hash value of the document as seen in the previous method is converted into base 64 encoding. The resulting value is then used to generate a QR code using ZXing library [15]. Figure 8 illustrates an example QR code of base 64 encoding of the document number 1 and SHA-256 of the title of this paper. When the user approves the document, the user takes the picture of the QR code with a mobile phone and the signature of the document is generated. The message containing the signature can be transmitted to the server via SMS or HTTP.

Figure 8. A QR code of base 64 encoding of the document number 1 and SHA-256 of the title of this paper.

The download method uses a HTTP connection to retrieve the document and generated the signature of the document. The document number, the certificate number, and the signature of the message are sent to the server via HTTP.

VI. Conclusion

This paper presents the design and analysis of three techniques for signing electronic documents using mobile phones. Electronic documents can be signed via SMS, QR codes, or downloads at the user's convenience depending on the available infrastructure. The security of each technique is analyzed. The SMS technique has some security concerns. However, the security can be enhanced by visually comparing the hash values. The QR code and download techniques provide strong security. With the use of mobile phones which are available to the majority of the people and are convenient, people will be more willing to use and sign electronic documents. Hence, they will use less paper.

References

[1] A.M. Yusoff and M.S. Sidhu, "Paperless in the electronic era: millennium dream and reality," in *Proc. 6th Conf. Computer Supported Cooperative Work in Design*, 2001, pp.536-541.

[2] K.L. Smart, "The paperless office: facts and fictions," in *Proc. IEEE Int. Professional Communication Conf.*, 1995. 1995, pp.141.

[3] A.J. Sellen and R.H.R. Harper, *The Myth of the Paperless Office*, Cambridge, MA: The MIT Press, 2003.

[4] *Measuring the Inform. Soc. 2010*, Int. Telecommun. Union, Geneva Switzerland, 2010.

[5] *Digital Signature Standard (DSS)*, FIPS PUB 186-3, 2009.

[6] *RSA Cryptography Standard*, PKCS #1 v2.1, 2002.

[7] *Public Key Cryptography for the Financial Services Industry, The Elliptic Curve Digital Signature Algorithm (ECDSA)*, ANSI X9.62-2005.

[8] *RSA Cryptography Standard*, PKCS #11 v2.20, 2004.

[9] About 2D Code. [Online]. Available: http://www.denso-wave.com/qrcode/aboutqr-e.html

[10] E. Ohbuchi et al., "Barcode readers using the camera device in mobile phones," in *Proc. Int. conf. on Cyberworlds*, 2004, pp. 260- 265.

[11] A. Sun et al., "The QR-code reorganization in illegible snapshots taken by mobile phones," in *Proc. Int. Conf. on Computational Sci. and its Applicat.*, 2007, pp.532-538.

[12] B. Dodson et al., "Secure, Consumer-Friendly Web Authentication and Payments with a Phone, in *Proc. 2nd Int. Conf. on Mobile Computing, Applications, and Services (MobiCASE)*, 2010.

[13] A. Hossain et al., "A Proposal for Enhancing The Security System of Short Message Service in GSM" in *Proc. 2nd Int. Conf. Anti-counterfeiting, Security and Identification, ASID*, 2008, pp. 235 – 240.

[14] Y. Wang, "Design of an Anti-counterfeiting System Based on SMS" in *Proc. IEEE Int. Conf. Granular Computing*, 2009, pp. 572-575.

[15] ZXing multi-format 1D/2D barcode image processing library, Available: http://code.google.com/p/zxing/